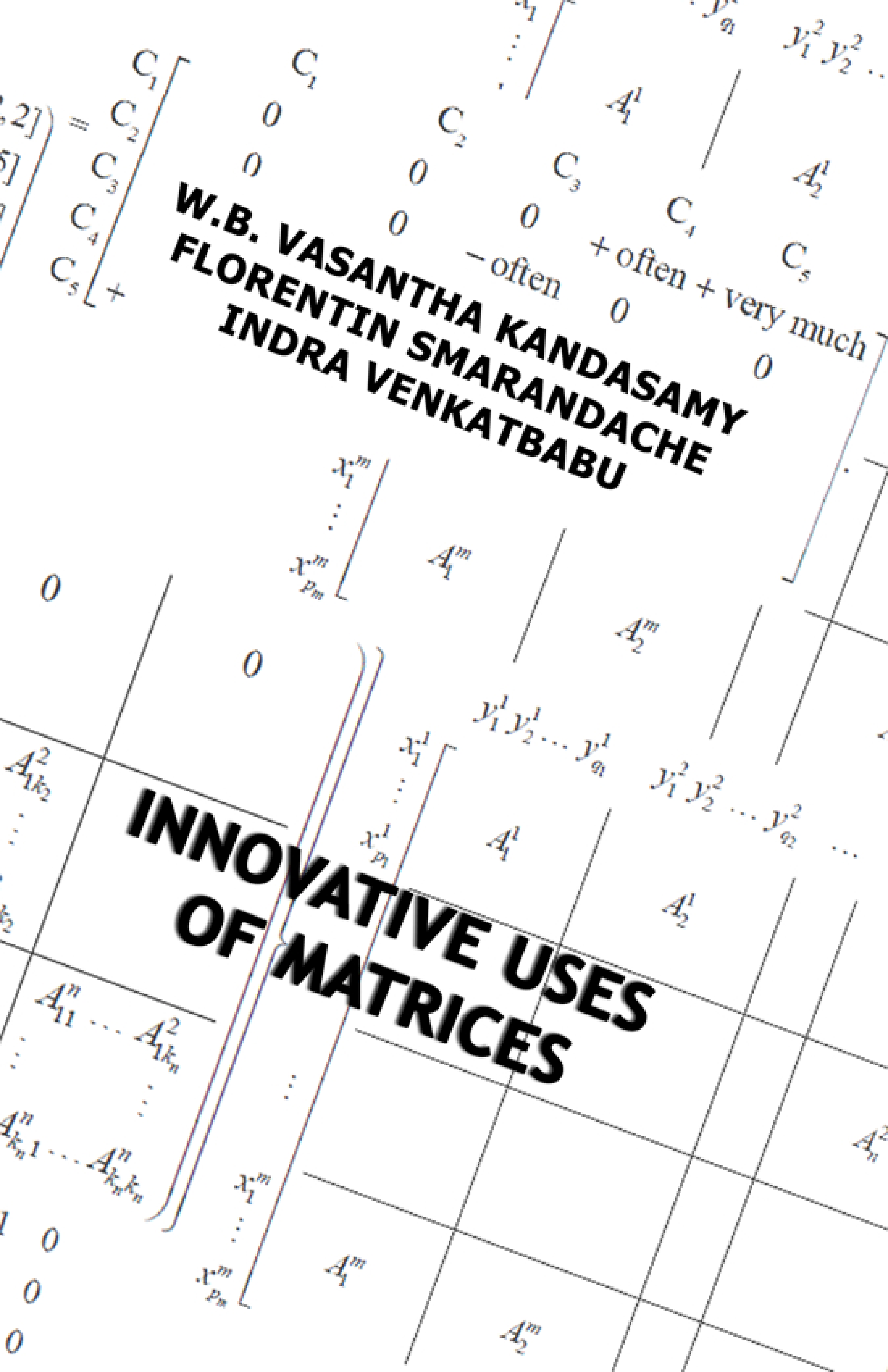


W.B. VASANTHA KANDASAMY
FLORENTIN SMARANDACHE
INDRA VENKATBABU

INNOVATIVE USES OF MATRICES

Innovative Uses of Matrices

W. B. Vasantha Kandasamy
Florentin Smarandache
Indra Venkatbabu

Educational Publisher Inc.
Ohio
2012

This book can be ordered from:

Education Publisher Inc.
1313 Chesapeake Ave.
Columbus, Ohio 43212,
USA
Toll Free: 1-866-880-5373

Peer reviewers:
Professor Paul P. Wang, Ph D, Department of Electrical & Computer Engineering, Pratt School of Engineering, Duke University, Durham, NC 27708, USA
Prof. Catalin Barbu, V. Alecsandri National College, Mathematics Department, Bacau, Romania
Prof. Gabriel Tica, Bailesti College, Bailesti, Jud. Dolj, Romania.

Many books can be downloaded from the following
Digital Library of Science:
http://fs.gallup.unm.edu/eBooks-otherformats.htm

ISBN-13: 978-1-59973-208-4
EAN: 9781599732084

Printed in the United States of America

CONTENTS

PREFACE

In this book authors bring out the innovative applications of matrices defined, described and developed by them. Here they do not include the natural product on matrices newly described and defined by them in the book on ‘natural product $\times_n$ on matrices’.

This book is organized into seven chapters. The first one is introductory in nature. In the second chapter authors give the unique and new way of analyzing the data which is time dependent. We construct three types of matrices called Average Time Dependent data matrix (ATD matrix), Refined Time Dependent Data matrix (RTD matrix) and Combined Effective Time Dependent Data matrix (CETD matrix). The authors describe the working of this new type of matrix model by working with the real world transportation problem. It is proved this new model is effective and elegant. At this juncture the authors deeply acknowledge the unostentatious help rendered by Dr. Mandalam.

In chapter three the authors for the first time define the new concept of matrices with linguistic variables. These linguistic matrices are used in the place of fuzzy matrices which serve as the dynamical system for FCM and FRM models. By this method the solution themselves are linguistic terms so mathematical interpretation is not needed.

In chapter four authors have used super matrices in the following ways. They are used in super fuzzy models like super FCMs, super FRMs super FAMs etc. Super linear algebra using these super matrices pave way to super eigen values and super eigen vectors, infact in almost all applications of matrices in linear algebra. The special mention is made to the Leontief open production super model and closed super model. By this method easy comparison and bulk working is possible.

We have also used super matrices with entries from finite fields in the construction of several types of super codes.

In chapter five we define interval matrix and matrix interval using the natural class of intervals, that is intervals of the form [a, b] where a < b or a > b or a and b not comparable. Using these interval matrices we have built interval linear algebra and these interval matrices when used as stiffness matrices of any mechanical problem easily yield a solution.

Suggestions and literature how we have used and constructed DSm matrices and DSm super matrices is given. However the authors have constructed DSm vector spaces and DSm super vector spaces. The eigen values and eigen vectors of these spaces will be refined labels and refined label vectors, which comprises chapter six of this book.

In the final chapter we have defined the new notion of bimatrices and n-matrices. Using these new concepts we have constructed linear algebra of type I and linear algebra of type II. Using these matrices we can build n-eigen values and n-eigen vectors.

Further these n-matrices are used in the construction of multi expert fuzzy models like n-FCMs, n-FRMs, n-FAMs etc. Finally we have built several types of n-codes using these n-matrices whose entries are from finite fields.

We thank Dr. K.Kandasamy for proof reading and being extremely supportive.

W.B.VASANTHA KANDASAMY
FLORENTIN SMARANDACHE
INDRA VENKATBABU

Chapter One

INTRODUCTION

This book gives the contributions of the authors in the innovative applications of matrices to various fields.

The authors were first to use matrix theory in the analysis of raw data. This is described in chapter two of this book. The method of applying matrices in this way makes one know the results and conclusions by just looking at it. Thus the final data is given a graph representation.

Secondly the authors have constructed matrices where entries are just linguistic terms which are orderable. For more about this refer [53]. Using these linguistic values we construct matrices. Infact these linguistic matrices are used in studying fuzzy linguistic models [53]. Thus we have in the first place constructed linguistic matrices and used them in mathematical models to study socio economic problems.

The authors have used super matrices in the first place to build super linear algebras. Almost all classical results of linear algebras are derived for the super linear algebras [45]. Using these super matrices we have built several types of super fuzzy models and these super fuzzy models can yield the result in a shorter time and comparision of the results are ready made [48].

Further these matrices are used in the construction of algebraic super codes of different types. For more about supercodes please refer [50].

We build matrix of refined labels and using these matrices we construct DSm vector spaces of refined labels. Several properties about DSm vector spaces build using matrix of refined labels in elaborately discussed. This is yet another means of applying matrix theory to DSm vector spaces of refined labels [46].

We also construct DSm super vector space of refined labels. For these new structures the notion of supermatrices is used. In the supermatrix if we replace the real entries or any other entries by refined labels L we call them as supermatrix of refined labels and also they form vector spaces. This is carried out in chapter six of this book [52].

Finally we build a natural class of intervals $\{[a, b] \mid a > b$ $a < b$ or $a = b$, $a, b \in Q$ or C or Z or $C(Z_n)$, Z_n or R$\}$. Using these natural classes of intervals we build interval matrices. These interval matrices behave like ordinary matrices as the entries are from this special natural class of intervals. Further study of stiffness matrix, load vector, mass matrix and damping matrix using interval for in this case lower and upper bound matrices is based on interval algebra. Also we find for square interval matrices, interval eigen values and interval eigen vectors [55].

Finally we define the notion of bimatrices and n-matrices. This is yet another innovative method of applying matrices. These n-matrices ($n \geq 2$) can be used in the construction of fuzzy n-models like n-FCMs, n-FRMs, n-BAMs etc., and their mixed structures [54, 56]. Finally we use these n-matrices in the construction of different types of n-codes [49].

Chapter Two

AVERAGE TIME DEPENDENT (ATD) DATA MATRIX

The raw data under investigation is classified under four broad heads viz; total number of passengers, total collection, number of trips, and hourly occupancy. These four broad heads form the columns of the matrices. The time periods of the day are represented by the rows of the matrices. Estimating the utility rate of a route is a five-stage process. In the first stage, we give a matrix representation of the raw data. Entries corresponding to the intersection of rows and columns are values corresponding to a live network. The initial $M \times N$ matrix is not uniform i.e., the number of individual hours in each time period may not be the same. So, in the second stage, we in order to obtain an unbiased uniform effect on each and every data so collected, transform this initial matrix into an Average Time Dependent Data matrix (ATD matrix). To make the calculations easier and simpler, we in the third stage, using the simple average techniques, convert the above time dependent data matrix into a matrix with entries e_{ij} where, $e_{ij} \in \{-1, 0, 1\}$.

We name this matrix as the Refined Time Dependent Data matrix (RTD matrix). The value of the e_{ij} corresponding to each entry is determined in a special way. At the fourth stage, using the refined time dependent data matrices, we get the Combined Effect Time Dependent Data matrix (CETD matrix) which gives the cumulative effect of all these entries. In the final stage, we obtain the row sums of the combined effect time dependent data matrix. We have written a Java program, which estimates all these five stages. [41]

Using the raw data available for any transport corporation, we analyze the raw data via matrices and (i) predict the maximum utilization time period (peak hours) of a day; and (ii) estimate the overall utility rate of the routes. Thus, to be more precise our chief problem here is prediction of the peak hours and estimation of the most utilized routes using the raw data available from any transport organization. We have established that the results which we have predicted using the raw data on the route 18R which ply from Parrys to Dharmaraja koil coincides with the peak hours and the utilization rate of the route which can be obtained from the estimation of the observed data. Thus, our analysis not only predicts the peak hours and the maximum utility routes but also estimates for each route, time periods where buses need not be operated as operation of buses in those time periods will result in total loss.

By total loss, we mean that the money collected from the passengers at that specified time period may not be even 20% of the cost of petrol and service charges spent on plying that service. So, even if the concern using the predictions on peak hours gets a profit of say 70%, if they do not comply with stopping of buses at the non-utilized hours, the transport corporation will still result only in loss. Hence, our approach via matrices not only suggests the peak hours and the maximum utilized routes but also predicts time periods and the routes in which the services should be totally avoided or reduced from their usual schedule in order to save themselves from loss. Thus, by adhering to our predictions the corporation might curtail the loss incurred by the operation of services in the non-utilized time periods.

The predictions can be explicitly seen even by a layman just by the observation of the combined effect time dependent data matrix mentioned earlier, which will be described and discussed in detail in the following.

Hence, using these results the Transport Corporation can operate more number of buses at the peak hours and on the routes that are highly utilized and also stop operating buses at odd hours or reduce the frequency at the odd hours of the day. Thus, the Transport Corporation can achieve a meager gain or curtail the loss by using these predictions.

We have considered the city of Madras, where the Pallavan Transport Corporation (PTC) is one of the transport organizations that caters to the demands of the community. The Pallavan Transport Corporation is a service oriented organization run by the state government for the people of the city and the neighbouring panchayats in the immediate vicinity of the city. It has a fleet of 1343 buses, and covers an overall of 283 routes and satisfies the needs of more than 75% of the urban transit commuters. Apart from the regular services, it has the partial services, the night services and the special services.

The regular services are of four types: normal, limited stop service, point-to-point service and the express service. Although the source, the destination and the path of travel is common to all the regular services, these services considerably differ in the travel time, the travel fare, the comfort and the convenience of a passenger. The basic difference observed in the above types of services is primarily due to the number of intermediate stops the vehicles halt in their course of travel.

The city transport corporation connects a vast network inter-linking various routes. The network has a set of nodes and a set of links connecting these nodes. Buses connect these nodes and ply over the links and help in passenger and freight transportation. In the city network, we observe that some routes have a high patronage almost throughout the day, while other routes are busy only at particular hours of a day and there are still some routes, which have only very meager passenger support throughout the day. Here, we not only predict the high utility and peak hours but also suggest the stoppage of meager passenger supported services.

To test the efficiency of the program, we have applied the algorithm to a few and established that the observed peak hours coincide well with the estimated peak hours for each route individually.

In the first stage of the problem, we split the total hours of a day into various time-periods viz. morning, morning peak, evening. Each of these time periods consists of individual hours of the day viz. $\{H_1, H_2, H_3, \ldots, H_{22}, H_{23}, H_{24}\}$, where H_i refers to the hour ending of the day. The working hours of the transport corporation under consideration is from $\{H_5, H_6, H_7, \ldots, H_{21}, H_{22}\}$. The data obtained from the transport corporation is for each hour ending. Hence, the time interval between H_i and H_{i+1} is 60 minutes. If we have all the related data viz. the total collection, the total number of passengers, the occupancy rate and the number of trips etc., for say every 30 minutes or for every 15 minutes, then the accuracy in the identification of the peak hours will further increase.

The finer the intervals are divided the better is the prediction. This is also established in this chapter. Each time period consists of a set of H_i individual hours, for example: the morning peak time period consists of $\{H_8, H_9, H_{10}\}$. These time-periods can vary from one analyst to another and from one route to another. Even, if we have each individual hour as a time period, still the total number of time periods must not exceed to total working hours of a day.

Let M represent the total number of time periods under consideration. The utility rate of a time period is influenced by the various attributes acting on them. Let N represent the number of attributes considered for the analysis. The various time periods are treated as rows and the various attributes are treated as columns of a matrix. The raw data corresponding to the intersection of each time period and attributes are treated as the entries of a matrix. Thus, from the raw data, we obtain the initial $M \times N$ matrix.

The number of individual hours in each time period might not be the same. Hence, in the second stage, in order to obtain consistency in the initial matrix we use the “hourly concept”. The entries corresponding to the intersection of each of the time periods and the attributes of the initial matrix are transformed so that each new entry corresponds to the hourly rate viz. total passengers per hour, total collection per hour, total number of trips per hour, total occupancy per hour etc.

Thus, we convert the above initial matrix into the Average Time Dependent Data matrix (ATD matrix) i.e., $[a_{ij}]_{M\times N}$.

In the third stage, we use the average and the standard deviation to convert to above average time dependent data matrix into a matrix with entries e_{ij}, $e_{ij} \in \{-1, 0, 1\}$, where i represents the i^{th} row and j represents the j^{th} column. We call this newly formed matrix as the Refined Time Dependent Data matrix (RTD matrix) i.e., $[e_{ij}]_{M\times N}$.

The value of the entry e_{ij} corresponding to each intersection is determined from an interval. This interval is obtained strictly by using the average and the standard deviation calculated from the raw data. The choice of the interval made by us might not be able to identify the accurate peak hour of the route; hence, we introduce and define a parameter α, which enables us to get a better solution.

We calculate the mean μ_j and the standard deviation σ_j for each attribute j, j = 1, 2, …. N using the data given in the average time dependent data matrix. For varying values of the parameter α, where $\alpha \in [0, 1]$. We follow a rule and determine the value of the entry e_{ij} in the refined time dependent data matrix.

For each of the attributes j (j = 1, 2, …, N) we have the rule:

If $a_{ij} \leq (\mu_j - \alpha * \sigma_j)$ then $e_{ij} = -1$;
else,
 if $a_{ij} \in (\mu_j - \alpha * \sigma_j, \mu_j + \alpha * \sigma_j)$ then $e_{ij} = 0$;
else
 if $a_{ij} \geq (\mu_j + \alpha * \sigma_j)$ then $e_{ij} = 1$;

here '*' denotes the usual multiplication.

Thus, for different values of α, we obtain different refined time dependent data matrices. The main purpose of introducing the refined time dependent data matrix is only to minimize the time involved in performing the simple arithmetic calculations and operations on the matrix.

In the fourth stage, we bring in the notion of the Combined Effect Time Dependent Data matrix (CETD matrix) i.e., $[c_{ij}]_{M \times N}$, which gives the combined effect of all the refined time dependent data matrices obtained by varying the parameter α. In the final stage, we add up the rows of the combined effect time-dependent data matrix. The overall time period utility of a route is obtained by inferring the row sums of a combined effect time-dependent data matrix. The highest positive value is taken as the highly utilized time period of a route and next lower value is taken as the next peak hour for the same route. Thus, for a particular route, we grade the utilization rate of the different time periods.

The computation starts by computing the ATD matrix from the initial $M \times N$ matrix. Then mean and standard deviations are computed for the ATD vector. Based on mean, standard deviation and ATD vector, the RTD and CETD matrices are computed either physically or using a programme in Java or C++.

It is pertinent to mention here this technique of analysing raw data can be used in case of any problem which is time dependent. This is a very new way of applying matrix theory to find solution for a collected raw data.

Parameters play a vital role in estimating the peak hours of a route. This is clearly illustrated for the route 18JJ and the route 17D by varying the parameter α. First, consider the route 18JJ. We have the following:

The initial M × N matrix obtained from the raw data is as below:

$$\begin{bmatrix} 81.00 & 63.00 & 3.00 & 147.20 \\ 925.00 & 157.00 & 16.00 & 1394.55 \\ 1040.00 & 181.00 & 15.00 & 1585.50 \\ 1159.00 & 271.00 & 13.00 & 1916.05 \\ 1596.00 & 400.00 & 18.00 & 2615.30 \\ 1673.00 & 451.00 & 16.00 & 2889.60 \\ 322.00 & 160.00 & 6.00 & 522.50 \end{bmatrix}.$$

The corresponding average time dependent data matrix is

$$\begin{bmatrix} 40.50 & 31.50 & 1.50 & 73.60 \\ 308.33 & 52.33 & 5.33 & 464.85 \\ 346.67 & 60.33 & 5.00 & 528.50 \\ 386.33 & 90.33 & 4.33 & 638.68 \\ 532.00 & 133.33 & 6.00 & 871.77 \\ 557.67 & 150.33 & 5.33 & 963.20 \\ 161.00 & 80.00 & 3.00 & 261.25 \end{bmatrix}.$$

The refined time dependent data matrix corresponding to a single value of $\alpha = 0.1$ is

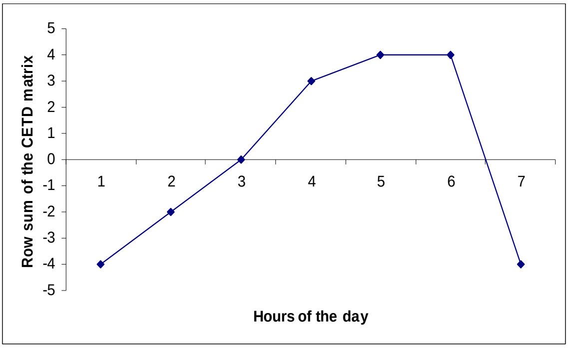
$$\begin{bmatrix} -1 & -1 & -1 & -1 \\ -1 & -1 & 1 & -1 \\ 0 & -1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 \end{bmatrix}$$

The corresponding row sum of the above refined time dependent data matrix is given by

Row sum for the first row is –4
Row sum for the second row is –2
Row sum for the third row is 0
Row sum for the fourth row is 3
Row sum for the fifth row is 4
Row sum for the sixth row is 4
Row sum for the seventh row is –4.

From the row sums of the above refined time dependent data matrix, we observe that time period corresponding to the fifth row i.e., time period $\{H_{16}, H_{17}, H_{18}\}$ and the sixth row i.e., time period $\{H_{19}, H_{20}, H_{21}\}$ are the peak hours of the route followed by the fourth row viz. time period $\{H_{13}, H_{14}, H_{15}\}$. Thus, graphically, we have this as below:

Graph depicting the peak periods of the Route 18JJ for $\alpha = 0.1$



The refined time dependent data matrix corresponding to the value of $\alpha = 0.4$ follows:

$$\begin{bmatrix} -1 & -1 & -1 & -1 \\ 0 & -1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ -1 & 0 & -1 & -1 \end{bmatrix}$$

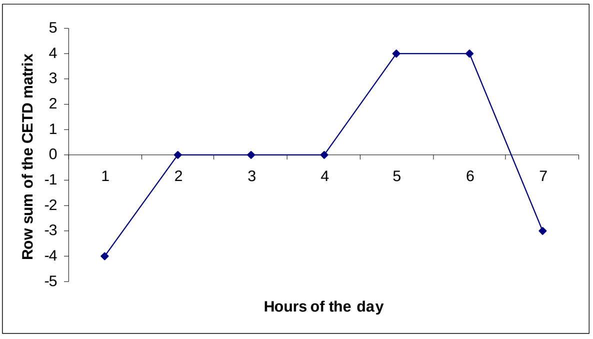
From the above, we have the row sums as given below:

Row sum for the first row is –4
Row sum for the second row is 0
Row sum for the third row is 0
Row sum for the fourth row is 0
Row sum for the fifth row is 4
Row sum for the sixth row is 4
Row sum for the seventh row is –3.

From the row sums of the above refined time dependent data matrix, we observe that time period corresponding to the fifth row i.e., time period $\{H_{16}, H_{17}, H_{18}\}$ and the sixth row i.e., time period $\{H_{19}, H_{20}, H_{21}\}$ are the peak hours of a day. Time periods $\{H_7, H_8, H_9\}$, $\{H_{10}, H_{11}, H_{12}\}$ and $\{H_{13}, H_{14}, H_{15}\}$ have a row sum of zero indicating that these time periods are neither the peak hours of a day nor the non-peak hours of a day. Hence, the number of trips made in these time periods should be kept unaltered for the concern to maintain the profit.

Thus, graphically, we have this as below:

Graph depicting the peak hour for the Route 18JJ for α = 0.4



The refined time dependent data matrix corresponding to the value of α = 0.7 follows:

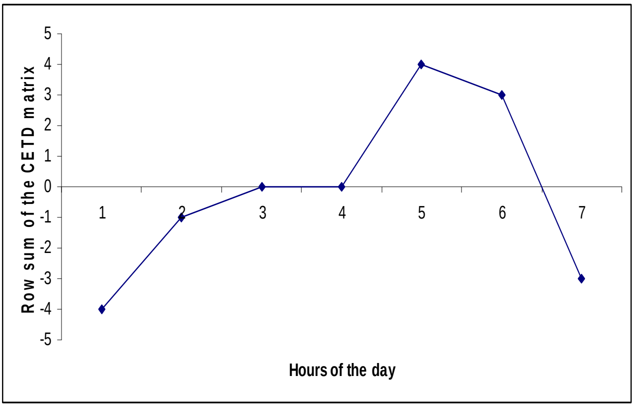
$$\begin{bmatrix} -1 & -1 & -1 & -1 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ -1 & 0 & -1 & -1 \end{bmatrix}$$

From the above, we have the row sums as given below:

Row sum for the first row is –4
Row sum for the second row is –1
Row sum for the third row is 0
Row sum for the fourth row is 0
Row sum for the fifth row is 4
Row sum for the sixth row is 3
Row sum for the seventh row is –3.

From the row sums of the above refined time dependent data matrix, we observe that time period corresponding to the fifth row i.e., time period $\{H_{16}, H_{17}, H_{18}\}$ is the first peak hour of the day followed by the sixth row i.e., time period $\{H_{19}, H_{20}, H_{21}\}$. Time periods $\{H_7, H_8, H_9\}$, $\{H_{10}, H_{11}, H_{12}\}$ and $\{H_{13}, H_{14}, H_{15}\}$ have a row sum of zero indicating that these time periods are neither the peak hours of a day nor the non-peak hours of a day. Thus, graphically, we have this as below:

Graph depicting the peak hour for the route 18 JJ for $\alpha = 0.7$



The refined time dependent data matrix corresponding to the value of $\alpha = 0.9$ follows:

$$\begin{bmatrix} -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ -1 & 0 & -1 & -1 \end{bmatrix}$$

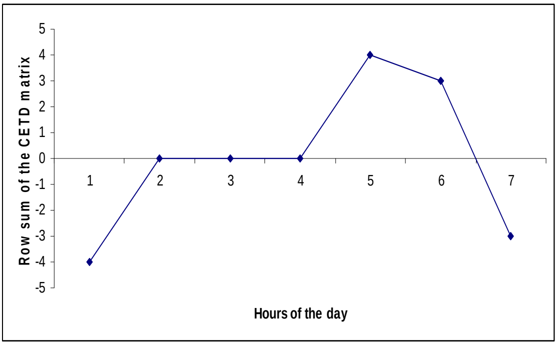
From the above, we have the row sums as given below:

Row sum for the first row is –4
Row sum for the second row is 0
Row sum for the third row is 0
Row sum for the fourth row is 0

Row sum for the fifth row is 4
Row sum for the sixth row is 3
Row sum for the seventh row is –3.

The row sums of the above matrix indicates time periods corresponding to the fifth row i.e., $\{H_{16}, H_{17}, H_{18}\}$ is the first peak hour of the day followed by the sixth row i.e., $\{H_{19}, H_{20}, H_{21}\}$. Time periods $\{H_7, H_8, H_9\}$, $\{H_{10}, H_{11}, H_{12}\}$ and $\{H_{13}, H_{14}, H_{15}\}$ have a row sum of zero indicating that these time periods are neither the peak hours of a day nor the non-peak hours of a day. Thus, graphically, we have this as below:

Graph depicting the peak hour for the route 18 JJ for α = 0.9



From the above analysis, we observe that the peak hours of a route vary from one time period to another with the change in the value of the parameter from 0 to 1. The row sums matrix obtained in the above cases were specific to only one value of the parameter. But the combined effect time dependent matrix for all the values of $\alpha \in [0, 1]$ is given below:

The combined time dependent data matrix for all $\alpha \in [0, 1]$ is

$$\begin{bmatrix} -10 & -10 & -10 & -10 \\ -1 & -8 & 6 & -2 \\ 0 & -6 & 4 & 0 \\ 3 & 1 & 0 & 3 \\ 10 & 10 & 10 & 10 \\ 10 & 10 & 6 & 10 \\ -9 & -1 & -9 & -9 \end{bmatrix}.$$

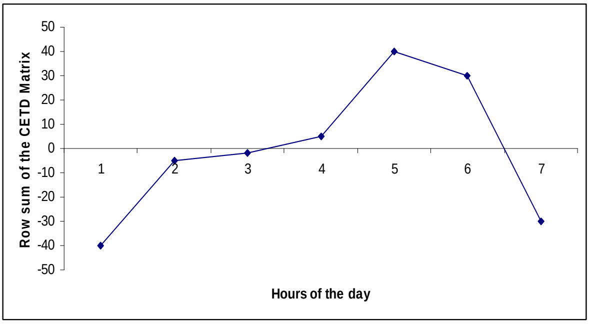
The row sum of the above combined time dependent data matrix is

Row = 1, row sum = –40
Row = 2, row sum = –5
Row = 3, row sum = –2
Row = 4, row sum = 7
Row = 5, row sum = 40
Row = 6, row sum = 36
Row = 7, row sum = –28.

From the row sums of the above matrix, we observe that the time period corresponding to the fifth row i.e., time period $\{H_{16}, H_{17}, H_{18}\}$ is the first peak hours of the day followed by the sixth row i.e., time period $\{H_{19}, H_{20}, H_{21}\}$ and the fourth row corresponding to the time period $\{H_{13}, H_{14}, H_{15}\}$. All the other time periods have a row sum of negative value indicating that these time periods are the non-peak hours of a day. The first time period has the least negative value indicating that services can be curtailed to the greatest extent in this time period.

Thus, graphically, we have the peak hour as depicted below: From the graph below, we observe that the peak hours as obtained from the combined time dependent data matrix, given by the cumulative effect of all the values of $\alpha \in [0, 1]$, gives the true peak hours of a day. These results coincide well with the observed hours of the route 18JJ.

Graph depicting the peak hour for the Route 18 JJ for $\alpha \in [0, 1]$



We give a grading of the peak hours since they are essential and helpful to the transportation sector in a number of ways:

1. It is efficient in identifying the highly utilized time periods along each route and enables the transport sector to operate more number of buses during these hours.

2. The transport sector identifies the poorly utilized time periods and makes necessary arrangements to overcome the loss incurred due to the services scheduled in these time periods. They might either discard all the trips made during these hours or operate minimum number of services.

3. The row sum of the combined time dependent data matrix of each route helps in identifying the overall utility rate of each route. In other words, if all the row sums of the combined time dependent data matrix are positive, then we conclude that the particular route is highly utilized throughout the day.

4. A positive value of the row sum of the combined time dependent data matrix indicates the maximum utilization of that specific time period.

5. A negative value of the row sums of the combined time dependent data matrix indicates that the passenger patronage and the collection in those time periods are very less. Hence, the transport sector can curtail the number of services totally, partially or operate mini buses to overcome the loss.

6. A zero value of the row sum of the combined time dependent data matrix indicates that the corresponding time periods are neither the peak hours of a day nor the non-peak hours of a day.

It is to be noted here that as the total number of time periods (M) increases, the accuracy in identifying the utilization rate of the different time-periods also increases. We substantiate our claim by taking two different values for the total number of time periods and this is carried out in the illustration I.

Illustration I (Route 18R)
The number of time periods is five (M = 5).

Here, we take five time periods viz. early morning C_1:{H_6, H_7}; morning C_2:{H_8, H_9, H_{10}, H_{11}}; noon C_3: {H_{12}, H_{13}, H_{14}, H_{15}}; evening C_4:{H_{16}, H_{17}, H_{18}, H_{19}}; and night C_5:{H_{20}, H_{21}, H_{22}}. We obtain the row sum of the combined time dependent data matrix as follows:

The combined time dependent data matrix corresponding to M = 5 is given below:

$$\begin{bmatrix} -10 & -8 & -10 & -10 \\ 6 & -2 & 6 & 5 \\ 2 & -1 & 6 & 1 \\ 10 & 10 & 6 & 10 \\ -4 & -6 & 0 & -3 \end{bmatrix}.$$

The row sums of the corresponding combined time dependent data matrices is given by:

Row sum for the time period early morning = –38
Row sum for the time period morning = 15
Row sum for the time period noon = 8
Row sum for the time period evening = 36
Row sum for the time period night = –13

The number of time periods is seventeen (M = 17).

Here, we take 17 time periods viz. early morning C_1 : {H_6}; morning C_2: {H_7}; C_3 : {H_8}, C_4 : {H_9}; early noon C_5: {H_{10}}; C_6: {H_{11}}, C_7: {H_{12}}; noon C_8: {H_{13}}; C_9: {H_{14}}, C_{10}: {H_{15}};

evening C_{11};{H_{16}}; C_{12}: {H_{17}}, C_{13} : {H_{18}}; late evening C_{14}: {H_{19}}, C_{15}:{H_{20}}, C_{16}:{H_{21}}; and night C_{17}: {H_{22}}. The row sums of the combined time dependent data matrices are obtained as follows:

The combined time dependent data matrix corresponding to M = 17 is as follows:

$$\begin{bmatrix} -10 & -2 & -10 & -10 \\ -10 & -7 & 0 & -10 \\ 0 & -5 & 0 & -5 \\ 4 & -2 & 0 & 2 \\ 7 & -1 & 0 & 5 \\ 0 & -2 & 0 & 0 \\ 1 & -3 & 10 & 2 \\ 3 & 0 & 0 & 2 \\ 2 & -1 & 0 & 0 \\ -10 & -2 & -10 & -10 \\ 10 & 1 & 10 & 10 \\ 6 & -1 & 0 & 6 \\ 10 & 3 & 0 & 10 \\ -4 & 10 & -10 & -4 \\ 10 & 0 & 10 & 10 \\ -4 & -2 & 0 & -3 \\ -10 & -7 & 0 & -10 \end{bmatrix}$$

The corresponding row sum of the above combined time dependent data matrix is given as below:

Row sum for the time period C_1:{H_6} = –32
Row sum for the time period C_2: {H_7} = –27
Row sum for the time period C_3: {H_8} = –10
Row sum for the time period C_4: {H_9} = 4

Row sum for the time period C_5:{H_{10}} = 11
Row sum for the time period C_6: {H_{11}} = –2
Row sum for the time period C_7: {H_{12}} = 10
Row sum for the time period C_8: {H_{13}} = 5

Row sum for the time period C_9:{H_{14}} = 1
Row sum for the time period C_{10}: {H_{15}} = –32
Row sum for the time period C_{11}: {H_{16}} = 31
Row sum for the time period C_{12}: {H_{17}} = 11

Row sum for the time period C_{13}:{H_{18}} = 23
Row sum for the time period C_{14}: {H_{19}} = –8
Row sum for the time period C_{15}: {H_{20}} = 30
Row sum for the time period C_{16}: {H_{21}} = –9
Row sum for the time period C_{17}: {H_{22}} = –27.

From the above Illustration I, on the analysis of the row sums of the combined time dependent data matrices of M = 5, we notice that the time period corresponding to row = 4 namely the evening hours {H_{16}, H_{17}, H_{18}, H_{19}} are the busy hours followed by row = 2 namely the morning hours {H_8, H_9, H_{10}, H_{11}}.

On analyzing the row sums of the combined time dependent data matrices of M = 17, we observe that row = 11 viz. evening {H_{16}} corresponds to the busiest hour of the route followed by row = 15 viz. late evening {H_{20}}, row = 13 viz. evening {H_{18}}.

Thus, we observe that as the number of time periods increases, the accuracy in identifying the busier hours of a route also increases. Also, we can observe the added effects of one or more attributes on the various time periods.

Thus, for $\alpha \in [0, 1]$ with a step function of $\alpha = 0.1$, the peak hours of the route are H_{16}, H_{20} and H_{18}. Time period corresponding to H_6 is poorly utilized and hence can be discarded.

The same problem is analyzed and the combined time dependent data matrix for varying values of $\alpha \in [0, 1]$ with a step function of $\alpha = 0.05$ is given below:

The combined time dependent data matrix, the row sum and the corresponding graph is given as below:

The combined time dependent data matrix

$$\begin{bmatrix} -10 & -4 & -10 & -10 \\ -10 & -10 & 0 & -10 \\ -1 & -10 & 0 & -10 \\ 8 & -4 & 0 & 5 \\ 10 & -3 & 0 & 10 \\ -1 & -5 & 0 & 1 \\ 2 & -7 & 10 & 5 \\ 7 & -1 & 0 & 4 \\ 4 & -3 & 0 & 1 \\ -10 & -4 & -10 & -10 \\ 10 & 2 & 10 & 10 \\ 10 & -2 & 0 & 10 \\ 10 & 7 & 0 & 10 \\ -8 & 10 & -10 & -9 \\ 10 & -1 & 10 & 10 \\ -9 & -5 & 0 & -7 \\ -10 & -10 & 0 & -10 \end{bmatrix}.$$

The row sums of the above combined time dependent data matrix is

Row = 1, row sum = –34
Row = 2, row sum = –30
Row = 3, row sum = –21
Row = 4, row sum = 9

Row = 5, row sum = 17
Row = 6, row sum = –5
Row = 7, row sum = 10
Row = 8, row sum = 10

Row = 9, row sum = 2
Row = 10, row sum = –34
Row = 11, row sum = 32
Row = 12, row sum = 18

Row = 13, row sum = 27
Row = 14, row sum = –17
Row = 15, row sum = 29
Row = 16, row sum = –21
Row = 17, row sum = –30.

We illustrate and describe the problem using 18R service.

We have applied our algorithm to a single route and obtained a comparison of the observed and the calculated data. We have taken the route 18R plying from Parrys to Dharmaraja Koil and the data was obtained from the Pallavan Transport Corporation Limited, Madras, India.

Here, we take 17 time periods viz. early morning C_1:{H_6}; morning C_2 : {H_7}; C_3:{H_8}, C_4 : {H_9}; early noon C_5 : {H_{10}}; C_6 : {H_{11}}, C_7 : {H_{12}}; noon C_8 : {H_{13}}; C_9:{H_{14}}, C_{10} : {H_{15}}; evening C_{11} : {H_{16}}; C_{12} : {H_{17}}, C_{13} : {H_{18}}; late evening C_{14} : {H_{19}}, C_{15} : {H_{20}}, C_{16} : {H_{21}}; and night C_{17} : {H_{22}} and also four attributes viz. A_1 : total population; A_2 : hourly occupancy; A_3 : number of trips and A_4 : total collection.

We view this problem from the aspect of both the operators as well as the users.

The initial M × N Matrix is as follows:

$$\begin{bmatrix} 96.00 & 64.00 & 1.00 & 156.95 \\ 71.00 & 21.00 & 2.00 & 103.55 \\ 222.00 & 39.00 & 2.00 & 244.90 \\ 269.00 & 63.00 & 2.00 & 353.50 \\ 300.00 & 70.00 & 2.00 & 392.80 \\ 220.00 & 60.00 & 2.00 & 328.30 \\ 241.00 & 53.00 & 3.00 & 353.65 \\ 265.00 & 78.00 & 2.00 & 348.65 \\ 249.00 & 70.00 & 2.00 & 327.85 \\ 114.00 & 63.00 & 1.00 & 151.60 \\ 381.00 & 91.00 & 3.00 & 526.40 \\ 288.00 & 73.00 & 2.00 & 407.05 \\ 356.00 & 112.00 & 2.00 & 515.15 \\ 189.00 & 389.00 & 1.00 & 252.75 \\ 376.00 & 78.00 & 3.00 & 569.40 \\ 182.00 & 59.00 & 2.00 & 261.90 \\ 67.00 & 20.00 & 2.00 & 81.35 \end{bmatrix}.$$

The average time dependent data matrix is as follows:

$$\begin{bmatrix} 96.00 & 64.00 & 1.00 & 156.95 \\ 71.00 & 21.00 & 2.00 & 103.55 \\ 222.00 & 39.00 & 2.00 & 244.90 \\ 269.00 & 63.00 & 2.00 & 353.50 \\ 300.00 & 70.00 & 2.00 & 392.80 \\ 220.00 & 60.00 & 2.00 & 328.30 \\ 241.00 & 53.00 & 3.00 & 353.65 \\ 265.00 & 78.00 & 2.00 & 348.65 \\ 249.00 & 70.00 & 2.00 & 327.85 \\ 114.00 & 63.00 & 1.00 & 151.60 \\ 381.00 & 91.00 & 3.00 & 526.40 \\ 288.00 & 73.00 & 2.00 & 407.05 \\ 356.00 & 112.00 & 2.00 & 515.15 \\ 189.00 & 389.00 & 1.00 & 252.75 \\ 376.00 & 78.00 & 3.00 & 569.40 \\ 182.00 & 59.00 & 2.00 & 261.90 \\ 67.00 & 20.00 & 2.00 & 81.35 \end{bmatrix}.$$

The refined time dependent data matrices corresponding to different values of alpha are calculated and from them we obtain the combined time dependent data matrix.

Since, the refined time dependent data matrices are obtained from the average time dependent data matrices they have a size equivalent to that of the average time dependent data matrices but, they have entries only in {–1, 0, 1}. For different values of the parameter we have different refined time dependent data matrices.

The refined time dependent data matrix corresponding to the value of $\alpha = 0.1$ is

–1	–1	–1	–1
–1	–1	0	–1
0	–1	0	–1
1	–1	0	1
1	–1	0	1
0	–1	0	0
1	–1	1	1
1	0	0	1
1	–1	0	0
–1	–1	–1	–1
1	1	1	1
1	–1	0	1
1	1	0	1
–1	1	–1	–1
1	0	1	1
–1	–1	0	–1
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value of $\alpha = 0.2$ is

–1	–1	–1	–1
–1	–1	0	–1
0	–1	0	–1
1	–1	0	1
1	0	0	1
0	–1	0	0
0	–1	1	1
1	0	0	1
1	0	0	0
–1	–1	–1	–1

1	0	1	1
1	0	0	1
1	1	0	1
–1	1	–1	–1
1	0	1	1
–1	–1	0	–1
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value of $\alpha = 0.3$ is

–1	0	–1	–1
–1	–1	0	–1
0	–1	0	–1
1	0	0	0
1	0	0	1
0	0	0	0
0	–1	1	0
1	0	0	0
0	0	0	0
–1	0	–1	–1
1	0	1	1
1	0	0	1
1	1	0	1
–1	1	–1	–1
1	0	1	1
–1	0	0	–1
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value α = 0.4 is

–1	0	–1	–1
–1	–1	0	–1
0	–1	0	–1
1	0	0	0
1	0	0	1
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1
1	0	1	1
1	0	0	1
1	0	0	1
–1	1	–1	–1
1	0	1	1
–1	0	0	0
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value α = 0.5 is

–1	0	–1	–1
–1	–1	0	–1
0	–1	0	–1
0	0	0	0
1	0	0	1
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1

1	0	1	1
1	0	0	1
1	0	0	1
0	1	–1	0
1	0	1	1
0	0	0	0
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value of $\alpha = 0.6$ is

–1	0	–1	–1
–1	–1	0	–1
0	0	0	0
0	0	0	0
1	0	0	0
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1
1	0	1	1
1	0	0	1
1	0	0	1
0	1	–1	0
1	0	1	1
0	0	0	0
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value of α = 0.7 is

–1	0	–1	–1
–1	–1	0	–1
0	0	0	0
0	0	0	0
1	0	0	0
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1
1	0	1	1
0	0	0	0
1	0	0	1
0	1	–1	0
1	0	1	1
0	0	0	0
–1	–1	0	–1

The refined time dependent data matrix corresponding to the value of α = 0.8 is

–1	0	–1	–1
–1	0	0	–1
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1

1	0	1	1
0	0	0	0
1	0	0	1
0	1	–1	0
1	0	1	1
0	0	0	0
–1	0	0	–1

The refined time dependent data matrix corresponding to the value of $\alpha = 0.9$ is

–1	0	–1	–1
–1	0	0	–1
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1
1	0	1	1
0	0	0	0
1	0	0	1
0	1	–1	0
1	0	1	1
0	0	0	0
–1	0	0	–1

The refined time dependent data matrix corresponding to the value of $\alpha = 1.00$ is

–1	0	–1	–1
–1	0	0	–1
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	1	0
0	0	0	0
0	0	0	0
–1	0	–1	–1
1	0	1	1
0	0	0	0
1	0	0	1
0	1	–1	0
1	0	1	1
0	0	0	0
–1	0	0	–1

The combined time dependent data matrix

$$\begin{bmatrix} -10 & -2 & -10 & -10 \\ -10 & -7 & 0 & -10 \\ 0 & -5 & 0 & -5 \\ 4 & -2 & 0 & 2 \\ 7 & -1 & 0 & 5 \\ 0 & -2 & 0 & 0 \\ 1 & -3 & 10 & 2 \\ 3 & 0 & 0 & 2 \\ 2 & -1 & 0 & 0 \\ -10 & -2 & -10 & -10 \\ 10 & 1 & 10 & 10 \\ 6 & -1 & 0 & 6 \\ 10 & 3 & 0 & 10 \\ -4 & 10 & -10 & -4 \\ 10 & 0 & 10 & 10 \\ -4 & -2 & 0 & -3 \\ -10 & -7 & 0 & -10 \end{bmatrix}.$$

The row sums corresponding to the above combined time dependent data matrix are:

Row sum for the time period C_1:{H_6} = –32
Row sum for the time period C_2:{H_7} = –27
Row sum for the time period C_3:{H_8} = –10
Row sum for the time period C_4:{H_9} = 4
Row sum for the time period C_5:{H_{10}} = 11

Row sum for the time period C_6:{H_{11}} = –2
Row sum for the time period C_7:{H_{12}} = 10

Row sum for the time period C_8:{H_{13}} = 5
Row sum for the time period C_9:{H_{14}} = 1

Row sum for the time period C_{10}:{H_{15}} = –32
Row sum for the time period C_{11}:{H_{16}} = 31

Row sum for the time period C_{12}:{H_{17}} = 11
Row sum for the time period C_{13}:{H_{18}} = 23
Row sum for the time period C_{14}:{H_{19}} = –8
Row sum for the time period C_{15}:{H_{20}} = 30
Row sum for the time period C_{16}:{H_{21}} = –9
Row sum for the time period C_{17}:{H_{22}} = –27

We observe that peak hours of a route differ for varying values of alpha. Hence, we form the combined effect time dependent data matrix which gives the combined effect of all the values of alpha in the interval [0, 1]. Thus, the highest value of the row for the corresponding time period identified from the row sums of the combined time dependent data matrix is observed as the best peak period of the particular route. The next value of the row sum in the small matrix is observed to be the next peak period of the same route. In this manner, we rank the different time periods of a route.

In the above estimated method, we notice that the time-period corresponding to row = 11 viz. evening {H_{16}} corresponds to the busiest hour of the route followed by row = 15 viz. late evening {H_{20}}, and row = 13 viz. evening {H_{18}}. Thus, we rank the peak periods for the route 18R to be as evening C_{11}:{H_{16}}; late evening C_{15}: {H_{20}} and evening C_{13} : {H_{18}} in the order of first, second, third respectively.

We now compare the predictions with the ridership method.

To establish the efficiency of our method, we compare our results with the ridership, approach followed by Dr. J.

Vasudevan in his research on the rationalization of city bus routes.

He had taken Madras as a case study where, he had discussed about the rationalization of the routes and had studied for a total of 90 routes. A small section of it deals with the average hourly bus passenger load for the city. For each hour ending, say 5 a.m., 6 a.m., …, 21 p.m., 22 p.m., he had calculated the total number of passengers who had traveled during that particular hour.

He had also calculated the percentage of the passenger load at any hour against the total strength of passengers who had traveled that day. This is clearly depicted by the following table.

Determination of the peak hour based on the Passenger load per hour

Hour ending	Passengers / hour	Percent to total
6	96	2.47
7	71	1.83
8	222	5.71
9	269	6.92
10	300	7.72
11	220	5.66
12	241	6.20
13	265	6.82
14	249	6.41
15	114	2.93
16	381	9.80
17	288	7.41
18	356	9.16
19	189	4.86
20	376	9.68
21	182	4.68
22	67	1.72
Total	3886	100

We have applied the matrix method to this data in order to obtain a comparison and establish that our method is better than the ridership method. From the last column of the above table, we identify the first peak period as the row corresponding to the greatest percent to the total, i.e., 9.80 corresponding to the hour ending 16, i.e., evening 4 p.m., followed by 9.68 corresponding to the hour ending 20 i.e., night 8 p.m.

The third peak period is the percent 9.16 corresponding to the hour ending 18 i.e., evening 6 p.m. In this fashion, we rank the peak periods of the route based on the passenger load.

Thus, we have established that our matrix method gives us a better result that the ridership method of Vasudevan.

Comparison Table depicting the peak hours as estimated by the three methods

Data as obtained from the transport corporation	**Identification of the peak and non-peak periods from the ridership method, the revenueship method and comparison of the two methods with the matrix method**		
Hour ending	**Observed Method I: Passengers /hour**	**Observed Method II: Revenue/hour**	**Matrix Method: row sum of the CETD Matrix**
6	96	156.95	–32
7	71	103.55	–27
8	222	244.9	–10
9	269	353.5	4
10	300	392.8	11
11	220	328.3	–2
12	241	353.65	10

13	265	348.65	5
14	249	327.85	1
15	114	151.6	–32
16	381(I)*	526.4 (II)*	31 (I)*
17	288	407.05	11
18	356 (III)*	515.15 (III)*	23 (III)*
19	189	252.75	–8
20	376 (II)*	569.4 (I)*	30 (II)*
21	182	261.9	–9
22	67	81.35	–27

*denotes the ranking of the time periods as given within parenthesis

From the above table, we obtain a comparative study of the observed methods and the estimated method. From the last column of the above table, we observe that there are rows that have a negative value.

These negative values indicate that the corresponding time periods are comparatively poor in performance i.e., in the passenger usage and in the collection. Hence, the services in these time periods can be discarded or the frequency of the services can be reduced.

This is one of the innovative means of applying matrices in analysis problems which has only raw data. Thus we have not given the graphs of them.

Chapter Three

FUZZY LINGUISTIC MATRICES AND THEIR APPLICATIONS

In this chapter we introduce the new notion of fuzzy linguistic matrices and use them in fuzzy linguistic models and apply them to social problems. We need for this the operations on the fuzzy linguistic matrices. We briefly describe the operations on them. Let M be any matrix if the entries of M are taken from the fuzzy linguistic set we call such matrix M to be a fuzzy linguistic matrix for more refer [53]. We describe operations on them. Let L denote the collection of fuzzy linguistic terms.

Let $x = (a_1, a_2, a_3, a_4, a_5)$ be a fuzzy linguistic row matrix (vector) with entries from L.

$$x^t = (a_1, a_2, a_3, a_4, a_5)^t = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix}$$

is the fuzzy linguistic column matrix (vector).

That is if x = (0, bad, good, fair, very bad) be the fuzzy linguistic row matrix, then transpose of x is denoted by

$$x^t = (0, \text{bad}, \text{good}, \text{fair}, \text{very bad})^t = \begin{bmatrix} 0 \\ \text{bad} \\ \text{good} \\ \text{fair} \\ \text{very bad} \end{bmatrix}$$

which is a fuzzy linguistic column matrix. Suppose we have product of a fuzzy linguistic row matrix x; it can be operated with min or max operation with a fuzzy linguistic column matrix y if and only if x is a $1 \times t$ row matrix then y must be a $t \times 1$ column matrix.

We show how the ‘min’ operation looks like.

Suppose

$$y = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_t \end{bmatrix} \text{ and } x = (b_1, b_2, \ldots, b_t)$$

then min {min {x, y}} = min {min $\{a_1, b_1\}$, min $\{a_2, b_2\}$, …, min $\{a_t, b_t\}$}.

Likewise we can have {max; max} operation; {max; min} operation and {min; max} operation. All these will be illustrated by examples.

Let $x = (x_1, x_2, \ldots, x_{20})$ and $y = (y_1, y_2, \ldots, y_{20})$ with $x_i, y_i \in L$; $1 \leq i \leq 20$. L be a fuzzy linguistic matrix.

max {max (x, y)} = max {max (x_1, y_1), max $\{x_2, y_2\}$, …, max $\{x_{20}, y_{20}\}$}.

Let x = (x_1, x_2, … x_{15}) and y = (y_1, y_2, …, y_{15}) max (min {x, y}) = max {min {x_1, y_1}, min {x_2, y_2}, …, min {x_{15}, y_{15}}}.

Likewise min (max {x, y}) = min {max {x_1, y_1}, max {x_2, y_2}, …, max {x_{15}, y_{15}}}.

We will see how for a typical problem the four operations give different sets of answers.

Let

$$x = \begin{bmatrix} \text{good} \\ \text{bad} \\ \text{fair} \\ 0 \\ \text{best} \\ \text{bad} \\ 0 \\ \text{good} \end{bmatrix} \text{ and}$$

y = (bad, 0, good, very bad, best, fair, best, better)

be two fuzzy linguistic matrices.

min {min (y, x)}

= min {bad, 0, fair, 0, best, bad, 0, better} = 0

max {min (y, x)}

= max {bad, 0, fair, 0, best, bad, 0, better} = best, min {max (y, z)}

= min {good, bad, good, very bad, best, fair, best, good}

= very bad and

max (max {y, x})

= max {good, bad, good, very bad, best, fair, best, good}

= best.

Suppose we are interested in finding min {x, y} then we have

min {x, y} =

$$\begin{bmatrix} \text{bad} & 0 & \text{good} & \text{very bad} & \text{good} & \text{fair} & \text{good} & \text{better} \\ \text{bad} & 0 & \text{bad} & \text{very bad} & \text{bad} & \text{bad} & \text{bad} & \text{bad} \\ \text{bad} & 0 & \text{fair} & \text{very bad} & \text{fair} & \text{fair} & \text{fair} & \text{fair} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \text{bad} & 0 & \text{good} & \text{very bad} & \text{best} & \text{fair} & \text{best} & \text{better} \\ \text{bad} & 0 & \text{bad} & \text{very bad} & \text{bad} & \text{bad} & \text{bad} & \text{bad} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \text{bad} & 0 & \text{good} & \text{very bad} & \text{good} & \text{fair} & \text{good} & \text{better} \end{bmatrix}.$$

This is the way a fuzzy linguistic column matrix of order 8 × 1 is multipliced with a 1 × 8 fuzzy linguistic row matrix with min operation gives a 8 × 8 fuzzy linguistic matrix.

We can also find max {x, y} =

$$\begin{bmatrix} \text{good} & \text{good} & \text{good} & \text{good} & \text{best} & \text{good} & \text{good} & \text{good} \\ \text{bad} & \text{bad} & \text{fair} & \text{bad} & \text{best} & \text{bad} & \text{bad} & \text{better} \\ \text{fair} & \text{fair} & \text{fair} & \text{fair} & \text{best} & \text{fair} & \text{fair} & \text{better} \\ \text{bad} & 0 & \text{fair} & 0 & \text{best} & \text{bad} & 0 & \text{better} \\ \text{best} & \text{best} & \text{best} & \text{best} & \text{best} & \text{best} & \text{best} & \text{best} \\ \text{bad} & \text{bad} & \text{fair} & \text{bad} & \text{best} & \text{bad} & \text{bad} & \text{better} \\ \text{bad} & 0 & \text{fair} & 0 & \text{best} & \text{bad} & 0 & \text{better} \\ \text{good} & \text{good} & \text{good} & \text{good} & \text{best} & \text{good} & \text{good} & \text{good} \end{bmatrix}.$$

We see clearly max {x, y} ≠ min {x, y}.

Further 0 dominates in min and best dominates in max. When an expert wants to boost the results on the positive side he can use the max operation. If min operation is used it gives the worst state of affairs.

Thus we see we get different results for these four types of operations. According to need one can use any one of the operations.

Now we find the transpose of a 6 × 5 fuzzy linguistic matrix M.

$$\text{Let } M = \begin{bmatrix} \text{good} & \text{best} & \text{bad} & \text{fair} & \text{best} \\ 0 & \text{bad} & \text{worst} & 0 & \text{very bad} \\ \text{bad} & \text{fair} & \text{very fair} & \text{bad} & \text{better} \\ \text{worst} & 0 & \text{good} & \text{better} & 0 \\ 0 & \text{fair} & 0 & \text{worst} & \text{best} \\ \text{good} & \text{fair} & \text{best} & \text{worst} & 0 \end{bmatrix}.$$

Now the transpose of this fuzzy linguistic 6 × 5 matrix M denoted

$$M^t = \begin{bmatrix} \text{good} & 0 & \text{bad} & \text{worst} & 0 & \text{good} \\ \text{best} & \text{bad} & \text{fair} & 0 & \text{fair} & \text{fair} \\ \text{bad} & \text{worst} & \text{very fair} & \text{good} & 0 & \text{best} \\ \text{fair} & 0 & \text{bad} & \text{better} & \text{worst} & \text{worst} \\ \text{best} & \text{very bad} & \text{better} & 0 & \text{best} & 0 \end{bmatrix}.$$

Likewise if

$$N = \begin{bmatrix} \text{best} & 0 & \text{bad} \\ \text{fair} & \text{bad} & 0 \\ 0 & \text{very fair} & \text{good} \\ \text{good} & 0 & \text{best} \\ \text{best} & \text{worst} & 0 \\ 0 & \text{best} & \text{good} \\ \text{good} & \text{better} & \text{best} \end{bmatrix}$$

is any fuzzy linguistic 7×3 matrix, to find N^T.

$$N^T = \begin{bmatrix} \text{best} & \text{fair} & 0 & \text{good} & \text{best} & 0 & \text{good} \\ 0 & \text{bad} & \text{very fair} & 0 & \text{worst} & \text{best} & \text{better} \\ \text{bad} & 0 & \text{good} & \text{best} & 0 & \text{good} & \text{best} \end{bmatrix}.$$

We see N^T is a 3×7 fuzzy linguistic matrix.

Now we can find the product of two rectangular linguistic matrices M and N if M is a $n \times t$ matrix then N must be a $t \times m$ matrix then only MN is defined how ever NM is not defined in this case and MN is a $n \times m$ fuzzy linguistic matrix.

However product of any $n \times n$ matrix with itself is always defined.

We will illustrate these two situations by some examples.

Let

$$M = \begin{bmatrix} \text{good} & \text{bad} & 0 & \text{worst} \\ \text{bad} & 0 & \text{good} & \text{best} \\ 0 & \text{good} & \text{bad} & 0 \\ \text{good} & \text{bad} & \text{good} & \text{bad} \end{bmatrix}$$

$$= \min(\min(M, M)) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

We see if ‘0’ occurs atleast once in every row and atleast once in every column then min (min (M, M)) = (0).

Now we find

$$\max(\min(M, M)) = \begin{bmatrix} \text{good} & \text{bad} & \text{bad} & \text{bad} \\ \text{good} & \text{good} & \text{good} & \text{bad} \\ \text{bad} & \text{bad} & \text{good} & \text{good} \\ \text{good} & \text{good} & \text{good} & \text{good} \end{bmatrix}.$$

We now find the value of

$$\max(\max(M, M)) = \begin{bmatrix} \text{good} & \text{good} & \text{good} & \text{best} \\ \text{best} & \text{best} & \text{best} & \text{best} \\ \text{good} & \text{good} & \text{good} & \text{best} \\ \text{good} & \text{good} & \text{good} & \text{best} \end{bmatrix}.$$

$$\min(\max(M, M) = \begin{bmatrix} 0 & \text{bad} & \text{bad} & 0 \\ \text{bad} & 0 & \text{bad} & \text{bad} \\ \text{bad} & \text{bad} & 0 & \text{bad} \\ \text{bad} & \text{bad} & \text{bad} & \text{bad} \end{bmatrix}.$$

We see max {max (M, M)} gives an extreme or better values and min {min {M, M}} gives an extreme low values, where as min max non negative values and max min more positive values.

Such four types of operations can be used as per need of the problem.

Now if M is a n × m matrix and N is a m × t matrix we can find min (min {M, N}), min (max {M, N}), max {max (M, N)} and max (min {M, N}).

We will illustrate these four types of operations.

Let

$$M = \begin{bmatrix} \text{low} & 0 & \text{high} & 0 & \text{very low} \\ 0 & \text{high} & 0 & \text{low} & \text{high} \\ \text{high} & 0 & \text{medium} & 0 & \text{low} \\ \text{low} & \text{low} & 0 & \text{medium} & 0 \end{bmatrix}$$

and

$$N = \begin{bmatrix} \text{low} & 0 & \text{medium} & 0 \\ \text{high} & \text{low} & 0 & \text{high} \\ 0 & \text{very high} & \text{low} & 0 \\ \text{medium} & 0 & \text{high} & \text{low} \\ \text{very low} & \text{medium} & 0 & \text{high} \end{bmatrix}$$

be two fuzzy linguistic matrices associated with temperature of an experiment.

$$\text{Clearly min (min } \{M, N\}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

$$\max \{\min (M, N)\} = \begin{bmatrix} \text{low} & \text{high} & \text{low} & \text{v.low} \\ \text{high} & \text{medium} & \text{low} & \text{high} \\ \text{low} & \text{medium} & \text{medium} & \text{low} \\ \text{medium} & \text{low} & \text{medium} & \text{low} \end{bmatrix}$$

$$\max\{\max (M, N)\} = \begin{bmatrix} \text{high} & \text{very high} & \text{high} & \text{high} \\ \text{high} & \text{very high} & \text{high} & \text{high} \\ \text{high} & \text{very high} & \text{high} & \text{high} \\ \text{high} & \text{very high} & \text{high} & \text{high} \end{bmatrix}.$$

$$\text{Now } \min \{\max (M, N)\} = \begin{bmatrix} \text{v.low} & 0 & 0 & \text{low} \\ 0 & 0 & \text{low} & 0 \\ \text{low} & 0 & 0 & \text{low} \\ 0 & \text{low} & 0 & 0 \end{bmatrix}.$$

We see this four types of operations gives four types of 4×4 fuzzy linguistic matrices.

Now we can also find linguistic row matrix with a square or a rectangular fuzzy linguistic matrix which is compatible.

Consider X = (fast, slow, very slow, just fast, very fast) to be the fuzzy linguistic row matrix.

Take

$$M = \begin{bmatrix} \text{slow} & \text{medium} & \text{fast} \\ \text{very slow} & \text{slow} & \text{medium} \\ \text{fast} & \text{slow} & \text{fast} \\ \text{fast} & \text{medium} & \text{slow} \\ \text{just fast} & \text{slow} & \text{very slow} \end{bmatrix}$$

be a 5 × 3 fuzzy linguistic matrix. Consider min {min {X, M}} = (very slow, very slow, very slow).

Consider max (min {X, M}) = (just fast, medium, fast).

Consider min (max {X, M}) = (slow, slow, medium).

To find max (max {X, M}) = (very fast, very fast, very fast).

Here we describe the notion of Fuzzy Linguistic Cognitive Models (FLCM).

For Fuzzy Linguistic Cognitive maps please refer [53].

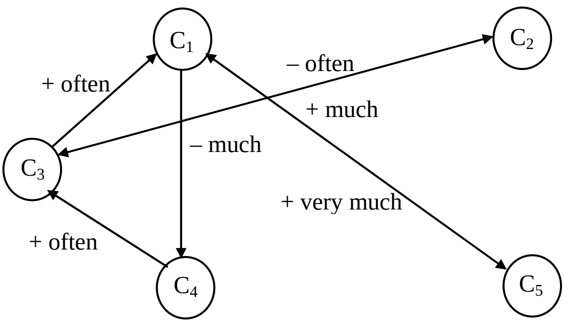
Example 3.1: Suppose we are interested in studying the child labour problem. Let (C_1, C_2, …, C_6) be six attributes / concepts associated with it.

C_1 - Child Labour
C_2 - Good Teacher
C_3 - School Drop out
C_4 - Poverty
C_5 - Public encouraging child labour

Let L = {0, often, + often, –often, very much, much, not that much, little, very little, more etc.}.

So the state vectors as well as the related fuzzy linguistic matrices take their values from the set L. Also the vertices of the fuzzy linguistic graph take their values from L.

Now using the experts opinion we have the following fuzzy linguistic graph.



Now using this linguistic graph we have the following fuzzy linguistic matrix M.

$$
M = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} C_1 & C_2 & C_3 & C_4 & C_5 \end{array} \\
\begin{bmatrix}
0 & 0 & 0 & +\text{ often} & +\text{ very much} \\
0 & 0 & -\text{ often} & 0 & 0 \\
+\text{ often} & -\text{ much} & 0 & 0 & 0 \\
0 & 0 & +\text{ often} & 0 & 0 \\
+\text{ very much} & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}.
$$

Now our state vector takes values from L. The min and max operations on L are defined as follows:

Let min $\{0, a_i\} = 0$ and max $\{0, a_i\} = a_i$ for all $a_i \in L$.
$a_i \in L$ min $\{a_i, a_i\} = a_i$ and max $\{a_i, a_i\} = a_i$.
min $\{a_i – a_i\} = – a_i$ and max $\{a_i, –a_i\} = a_i$ for all $a_i \in L$.
For instance min {often, very often} = often
max {often, very often} = very often
min {much, often} = often,
max {much, often} = much
min {–much, often} = – much
and max {–much, often} = often.
Like this operations on L are performed.

Now we find x_1M using as before min {min (a_i, m_{ij})} where $x_1 = (a_1, a_2, …, a_6)$ and $M = (m_{ij})$; $m_{ij}, a_i \in L$, $1 \leq i, j \leq 6$.

Thus x_1M = (+ often, 0, + often, + often, 0) ×

$$\begin{bmatrix} 0 & 0 & 0 & +\text{ often} & +\text{ very much} \\ 0 & 0 & -\text{ often} & 0 & 0 \\ +\text{ often} & -\text{ much} & 0 & 0 & 0 \\ 0 & 0 & +\text{ often} & 0 & 0 \\ +\text{ very much} & 0 & 0 & 0 & 0 \end{bmatrix}$$

= (+ often, 0, + often, + often, + often)
leading to a fixed point.

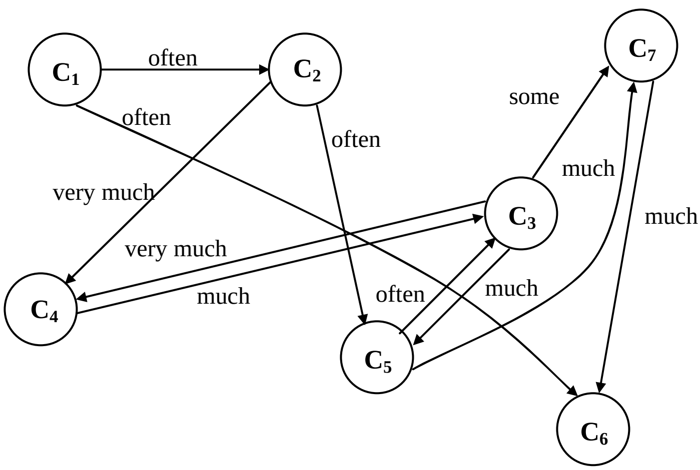
Example 3.2: Consider the problem of finding the seven transit system which includes the level of service and the convenience factors. We have the following eight attributes.

C_1 - Frequency of the service along a route
C_2 - In-vehicle travel time along the route
C_3 - Travel fare along the route
C_4 - Speed of the vehicles along the route

C_5 - Number of intermediate points in the route.
C_6 - Waiting time
C_7 - Number of transfers in the route

The fuzzy linguistic terms associated with $C_1, C_2, \ldots, C_7$ as well as the problem are L = {0, often, always, a little, much, very much, usually, some times}.

We now give the fuzzy linguistic graph whose vertices are $C_1, C_2, \ldots, C_7$ and the edges take values from L are as follows.



$$M = \begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \\ C_7 \end{array} \begin{array}{c} \begin{array}{ccccccc} C_1 & C_2 & C_3 & C_4 & C_5 & C_6 & C_7 \end{array} \\ \left[\begin{array}{ccccccc} 0 & \text{often} & 0 & 0 & 0 & \text{often} & 0 \\ 0 & 0 & 0 & \text{very much} & \text{often} & 0 & 0 \\ 0 & 0 & 0 & \text{very much} & \text{much} & 0 & \text{some} \\ 0 & 0 & \text{much} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{often} & 0 & 0 & 0 & \text{much} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{much} & 0 & 0 \end{array}\right] \end{array}.$$

Let X = (often, 0, some, 0, often, some, 0) is the fuzzy linguistic state vector.

To study the effect of X on M using max {max {X, M}} after updating max {max {X, M}} is (often, often, some, very much, often, some, much) = X_1.

Now max {max {X_1, M}} after updating is say X_2;

X_2 = (often, very much, some, very much, often, some, very much).

Now we find max {max (X_2, M)} after updating we get say X_3;

X_3 = (often, very much, some, very much, often, some, very much).

Now suppose another expert wants to use for the same fuzzy linguistic state vector X and the same dynamical system, to study the effect using the 'max min' operation.

Now max {min (X, M)} after updating we get say X_1,

X_1 =(often, often, some, 0, often, some, often).

We find max {min (X, M)} after updating we get say X_2. This is the way the operations are performed on the fuzzy linguistic matrix.

X_2 = (often, often, some, 0, often, some often).

For about Fuzzy linguistic relational map-model refer [53].

We can use any of the four types of operations {max, max} or {min, max} or {max, min} or {min, min} in these fuzzy linguistic matrices [53].

We will illustrate this situation by an example.

Example 3.3: Let us study the employee - employer fuzzy linguistic relational model. Suppose we have the following fuzzy linguistic concepts / attributes associated with the employee taken as the domain space

D_1 - Pay with allowances and bonus

D_2 - Only pay to employee

D_3 - Pay with allowance to employee

D_4 - Best performance by the employee

D_5 - Average performance by the employee

D_6 - Employee works for more number for hours.

Suppose the following nodes / concepts are taken as the range space of the employer.

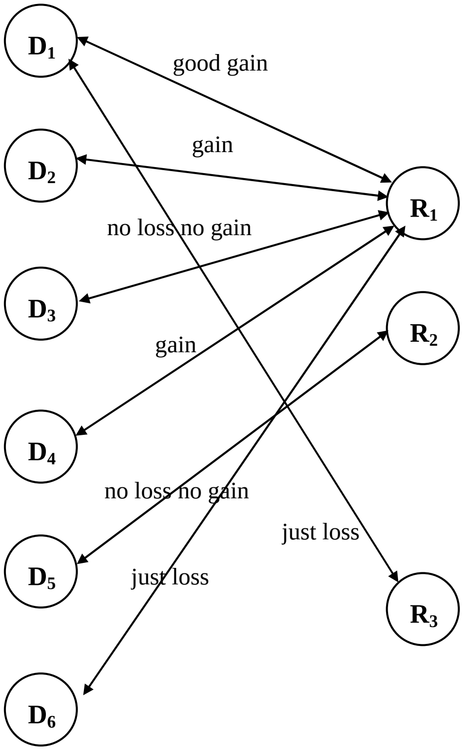
R_1 - maximum profit to the employer

R_2 - Only profit to the employer

R_3 - Neither profit nor less to the employer.

The fuzzy linguistic terms associated with the fuzzy linguistic domain and range spaces be taken as L.

L = {0, gain, loss, no loss no gain, just gain, just loss, gain, heavy loss, good gain}.



We give the associated fuzzy linguistic matrix N of the fuzzy linguistic graph.

$$N = \begin{array}{c} \\ D_1 \\ D_2 \\ D_3 \\ D_4 \\ D_5 \\ D_6 \end{array} \begin{array}{c} \begin{array}{ccc} R_1 & R_2 & R_3 \end{array} \\ \left[\begin{array}{ccc} \text{good gain} & 0 & \text{just loss} \\ \text{gain} & 0 & 0 \\ \text{no loss no gain} & 0 & 0 \\ \text{gain} & 0 & 0 \\ 0 & \text{no loss no gain} & 0 \\ \text{just loss} & 0 & 0 \end{array} \right] \end{array}.$$

Now we find the resultant of any fuzzy linguistic vector on N, the dynamical system associated with the problem.

Let X = (gain, 0, loss, 0, gain, loss) be the given fuzzy linguistic state vector. The effect of X on the fuzzy linguistic dynamical system N is as follows:

Max {max (X, N)} = ((good gain, gain, gain)} = Y

We find max (max {Y, N^T}) = (gain, gain, loss, gain, gain, loss) = X_1 (say) after updating.

We find max (max {X_1, N})

= (good gain, gain, gain) = Y_1 = Y.

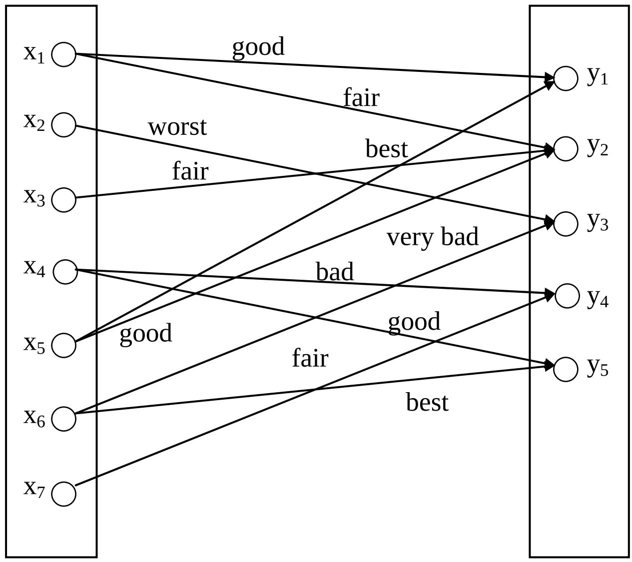
This is the way the operations using linguistic matrices are performed on these models.

Finally for the first time we introduce the notion of fuzzy linguistic relation equations and describe some of the properties related with them.

Let L_1 and L_2 be any two fuzzy linguistic sets that is both L_1 and L_2 contain fuzzy linguistic terms or $L_1 = L_2$ otherwise. Let R be a fuzzy linguistic relation that is to each fuzzy linguistic term of L_1 two or more fuzzy linguistic terms in L_2 are assigned, that is to each $x_1 \in L_1$ to the domain space (L_1 is the domain linguistic space) we associate a $y_2 \in L_2$ and here the degree of membership is not a value between [0, 1] but a fuzzy linguistic value from L.

This we will first illustrate by some example.

Let $L_1 = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ be seven fuzzy linguistic terms and $L_2 = \{y_1, y_2, y_3, y_4, y_5\}$ be some five fuzzy linguistic terms with the following relation.



We have fuzzy linguistic membership matrix R associated with the above map.

$$R = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{array} \begin{array}{c} \begin{array}{ccccc} y_1 & y_2 & y_3 & y_4 & y_5 \end{array} \\ \begin{bmatrix} \text{good} & \text{fair} & 0 & 0 & 0 \\ 0 & 0 & \text{worst} & 0 & 0 \\ 0 & \text{fair} & 0 & 0 & 0 \\ 0 & 0 & 0 & \text{bad} & \text{good} \\ \text{best} & \text{good} & 0 & 0 & 0 \\ 0 & 0 & \text{v.bad} & 0 & \text{best} \\ 0 & 0 & 0 & \text{fair} & 0 \end{bmatrix} \end{array}$$

Thus for fuzzy linguistic sets L_1 and L_2 we can have a fuzzy linguistic membership function R. The representation by the diagram will be known as the fuzzy linguistic sagittal diagram and the fuzzy linguistic matrix will be known as the fuzzy linguistic membership matrix where the membership values are also fuzzy linguistic terms.

We can compose two fuzzy linguistic binary relation provided both of them take their fuzzy linguistic membership values from the same fuzzy linguistic set L.

We will just illustrate how such compositions are made.

Suppose

$$P = \begin{bmatrix} \text{good} & \text{bad} & \text{fair} & \text{best} \\ \text{bad} & \text{fair} & \text{good} & \text{good} \\ 0 & \text{good} & \text{fair} & \text{good} \\ \text{good} & \text{bad} & \text{good} & \text{fair} \end{bmatrix}$$

and

$$Q = \begin{bmatrix} \text{good} & \text{bad} & \text{good} & \text{fair} & 0 & \text{best} \\ \text{bad} & \text{good} & \text{best} & \text{good} & \text{bad} & 0 \\ \text{best} & \text{fair} & \text{best} & 0 & \text{good} & \text{bad} \\ 0 & \text{fair} & \text{good} & \text{bad} & \text{good} & \text{good} \end{bmatrix}$$

be two fuzzy linguistic membership matrices of a fuzzy linguistic binary relation with the values taken from the same fuzzy linguistic space L.

Thus we have used the innovative methods of applying matrices by constructing in the first place fuzzy linguistic matrices and secondly using them in fuzzy models like Fuzzy Cognitive Maps, Fuzzy Relational Maps and Fuzzy Relational Equations.

Now

$$P \text{ o } Q = \begin{bmatrix} \text{good} & \text{bad} & \text{fair} & \text{best} \\ \text{bad} & \text{fair} & \text{good} & \text{good} \\ 0 & \text{good} & \text{fair} & \text{good} \\ \text{good} & \text{bad} & \text{good} & \text{fair} \end{bmatrix} \text{ o }$$

$$\begin{bmatrix} \text{good} & \text{bad} & \text{good} & \text{fair} & 0 & \text{best} \\ \text{bad} & \text{good} & \text{best} & \text{good} & \text{bad} & 0 \\ \text{best} & \text{fair} & \text{best} & 0 & \text{good} & \text{bad} \\ 0 & \text{fair} & \text{good} & \text{bad} & \text{good} & \text{good} \end{bmatrix} = R$$

where R is found in the following way.

$$\text{If } R = \begin{bmatrix} r_1 & r_2 & r_3 & r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 & r_{10} & r_{11} & r_{12} \\ r_{13} & r_{14} & r_{15} & r_{16} & r_{17} & r_{18} \\ r_{19} & r_{20} & r_{21} & r_{22} & r_{23} & r_{24} \end{bmatrix} \text{ then}$$

r_1 = max [min (good, good), min (bad, bad), min (fair, best), min (best, 0)]
= max [good, bad, fair, 0] = good

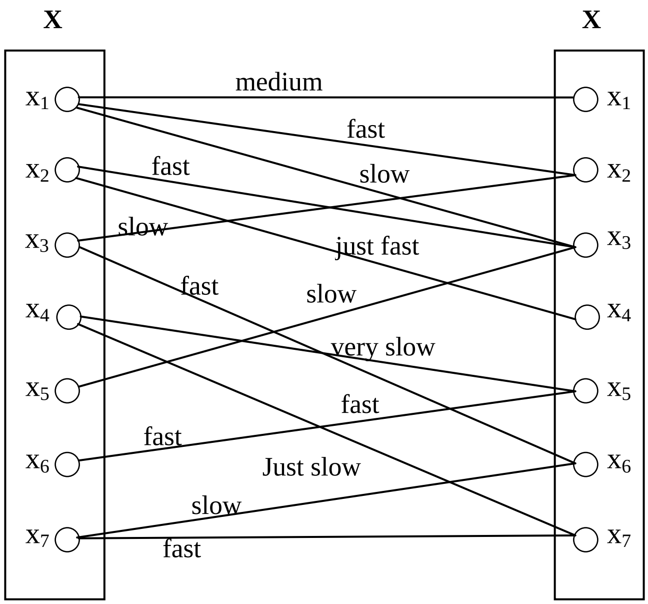
r_2 = max [min (good, bad), min (bad, good), min (fair, fair), min (best, fair)}
= max {bad, bad, fair, fair} = fair

r_3 = max {min {good, good}, min {bad, best}, min {fair, best}, min {best, good}}
= max {good, bad, fair, good} = good

r_4 = max {min {good, fair}, min {bad, good},
min {fair, 0}, min {best, bad}}
= max {fair, bad, 0, bad } = fair.

and so on.

We have the following fuzzy linguistic binary relation which is described by the fuzzy linguistic sagittal diagram and the related fuzzy linguistic membership matrix.



The fuzzy linguistic membership matrix associated with X is

$$\begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{array} \begin{array}{c} \begin{array}{ccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \end{array} \\ \left[\begin{array}{ccccccc} \text{medium} & \text{fast} & \text{slow} & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{fast} & \text{just fast} & 0 & 0 & 0 \\ 0 & \text{slow} & 0 & 0 & 0 & \text{fast} & 0 \\ 0 & 0 & 0 & 0 & \text{very slow} & 0 & \text{fast} \\ 0 & 0 & \text{slow} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \text{fast} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \text{slow} & \text{fast} \end{array}\right] \end{array}.$$

Operations can be performed on these matrices.

Chapter Four

SUPERMATRICES AND THEIR APPLICATIONS

The concept of super matrices was first introduced by [17]. Later the authors developed a new algebra using these matrices called super linear algebras [45]. Thus for the first time we use only super matrices to build super linear algebras. This application of super matrices is not only innovative and has lot of applications. Further one can find bulk eigen values using super square diagonal matrices. This method helps in easy comparison of eigen values and also bulk performances takes lesser time [45].

Apart from this super fuzzy matrices are used in the construction of super fuzzy models like New Super Fuzzy Relational model, New Super Fuzzy Cognitive Maps model, New Super Fuzzy Associative Maps model, we make use of them in the construction of super codes [50].

In this section for the first time we introduce the new notion of Super Fuzzy Relational Maps (SFRMs) models and they are applied to real world problems, which is suited for multi expert problems. When we in the place of fuzzy relational matrix of the FRM (using single expert) use fuzzy supermatrix with multi experts we call the model as Super Fuzzy Relational Maps (SFRMs) model.

We just recall the definition of Domain Super Fuzzy Relational Maps (DSFRMs) model.

DEFINITION 4.1: *Suppose we have some n experts working on a real world model and give their opinion. They all agree upon to work with the same domain space elements / attributes / concepts; using FRM model but do not concur on the attributes from the range space then we can use the special super fuzzy row vector to model the problem using Domain Super Fuzzy Relational Maps (DSFRMs) Model.*

The DSFRM matrix associated with this model will be given by S_M

$$S_M = \begin{array}{c} \\ D_1 \\ D_2 \\ \vdots \\ D_m \end{array} \begin{array}{c} \begin{array}{cccc} t_1^1 \ldots t_{r_1}^1 & t_1^2 \ldots t_{r_2}^2 & & t_1^n\, t_2^n \ldots t_{r_n}^n \end{array} \\ \left[\begin{array}{c|c|c|c} & & & \\ & & & \\ & & & \\ & & & \end{array} \right] \end{array}$$

$$= \left[S_M^1 \mid S_M^2 \mid \ldots \mid S_M^n \right]$$

where each S_M^i is a m × $t_{r_i}^i$ matrix associated with a FRM given by the i^{th} expert having D_1, ..., D_m to be the domain attributes and ($t_1^i\, t_2^i\, \ldots\, t_{r_i}^i$) to be the range attributes of the i^{th} expert, i = 1, 2, ..., n and S_M the DSFRM matrix will be a special super row vector / matrix ($1 \le i \le n$).

However if n is even a very large value using the mode of programming one can easily obtain the resultant vector or the super hidden pattern for any input supervector which is under investigation.

These DSFRMs will be known as Domain constant DSFRMs for all the experts choose to work with the same domain space attributes only the range space attributes are varying denoted by DSFRM models.

We see these super row vectors (super row matrices) find their applications in the DSFRMs model.

Next we proceed on to recall the notion of super FRMs with constant range space attributes and varying domain space attributes.

DEFINITION 4.2: *Let some m experts give opinion on a real world problem who agree upon to make use of the same space of attributes / concepts from the range space using FRMs but want to use different concepts for the domain space then we make use of the newly constructed special super column vector as the matrix to construct this new model.*

The column attributes i.e. the range space remain constant as $R_1, \ldots, R_s$ for all m experts only the row attributes for any i^{th} expert is $D_1^i, D_2^i, \ldots, D_{t_i}^i$; i = 1, 2, ..., m; vary from expert to expert. This system will be known as the Range constant fuzzy super FRM or shortly denoted as RSFRM model.

Thus the associated special super column matrix S^M is

$$S^M = \begin{array}{c} \\ D_1^1 \\ D_2^1 \\ \vdots \\ D_{t_1}^1 \\ D_1^2 \\ \vdots \\ D_{t_2}^2 \\ \\ \vdots \\ \\ D_1^m \\ \vdots \\ D_{t_m}^m \end{array} \begin{array}{c} R_1 \quad R_2 \quad \ldots \quad R_s \\ \left[\begin{array}{c} \\ \\ \\ \\ \hline \\ \\ \\ \hline \\ \\ \hline \\ \\ \\ \end{array}\right] \end{array}$$

These super row vectors (super row matrices) find applications in the RSFRM models.

DEFINITION 4.3: *Suppose we have m experts who wish to work with different sets of both row and column attributes i.e. domain and range space using FRMs, then to accommodate or form a integrated matrix model to cater to this need. We make use of the super diagonal fuzzy matrix, to model such a problem. Suppose the first expert works with the domain attributes* $D_1^1, \ldots, D_{t_1}^1$ *and range attributes* $R_1^1, \ldots, R_{n_1}^1$*, The second expert works with domain attributes* $D_1^2, \ldots, D_{t_2}^2$ *and with range attributes* $R_1^2, \ldots, R_{n_2}^2$ *and so on. Thus the* m^{th} *expert works with* $D_1^m, \ldots, D_{t_m}^m$ *domain attributes and* $R_1^m, \ldots, R_{n_m}^m$ *range attributes. We have the following diagonal fuzzy supermatrix to model the situation. We are under the assumption that all the attributes both from the domain space as well as the range space of the m experts are different.*

The super fuzzy matrix S associated with this new model is given by

$$
S = \begin{array}{c}
 \\
D_1^1 \\ \vdots \\ D_{t_1}^1 \\ D_1^2 \\ \vdots \\ D_{t_2}^2 \\ \vdots \\ D_1^m \\ \vdots \\ D_{t_m}^m
\end{array}
\begin{array}{c}
\begin{array}{cccc}
R_1^1\, R_2^1 \ldots R_{n_1}^1 & R_1^2\, R_2^2 \ldots R_{n_2}^2 & \ldots & R_1^m\, R_2^m \ldots R_{t_m}^m
\end{array} \\
\left[\begin{array}{c|c|c|c}
M_1 & (0) & & (0) \\ \hline
(0) & M_2 & & (0) \\ \hline
(0) & (0) & & (0) \\ \hline
(0) & (0) & & M_n
\end{array}\right]
\end{array}
$$

where each M_i is a $t_i \times n_i$ matrix associated with the FRM, we see except, the diagonal strip all other entries are zero.

We call this matrix as a special diagonal super fuzzy matrix and this model will be known as the Special Diagonal Super FRM Model which will be denoted briefly as (SDSFRM).

The super diagonal matrices are used in the SDSFRM models.

DEFINITION 4.4: *Suppose one is interested in finding a model where some mn number of experts work on the problem and some have both domain and range attributes to be not coinciding with any other expert and a set of experts have only the domain attributes to be in common and all the range attributes are different.*

Another set of experts are such that only the range attributes to be in common and all the domain attributes are different, and all of them wish to work with the FRM model only; then we model this problem using a super fuzzy matrix. We have mn experts working with the problem.

Let the t_1 expert wish to work with domain attributes P_1^1, P_2^1, …, $P_{m(t_1)}^1$ *and range attributes* q_1^1, q_2^1, …, $q_{n(t_1)}^1$.

The t_2 expert works with P_1^1, P_2^1, …, $P_{m(t_1)}^1$ *as domain attributes and the range attributes* q_1^2, q_2^2, …, $q_{n(t_2)}^2$ *and so on. Thus for the t_i expert works with* P_1^i, P_2^i, …, $P_{m(t_i)}^i$ *as domain space attributes and* q_1^i, q_2^i, …, $q_{n(t_i)}^i$, *as range attributes ($1 \leq i \leq m(t_i)$ and $i \leq n(t_i)$).*

So with these mn experts we have an associated super FRM matrix. Thus the supermatrix associated with the Super FRM (SFRM) model is a supermatrix of the form

$$S(m) = \left[\begin{array}{c|c|c|c} A^{11}_{m(t1)n(t1)} & A^{12}_{m(t1)n(t2)} & & A^{1n}_{m(t1)n(tn)} \\ \hline A^{21}_{m(t2)n(t1)} & A^{22}_{m(t2)n(t2)} & & A^{2n}_{m(t2)n(tn)} \\ \hline & & & \\ \hline A^{m1}_{m(tm)n(t1)} & A^{m2}_{m(tm)n(t2)} & & A^{mn}_{m(tm)n(tn)} \end{array}\right]$$

where

$$A^{ij}_{m(t_i)n(t_j)} = \begin{array}{c} \\ P^i_1 \\ P^i_2 \\ \vdots \\ P^i_{m(t_i)} \end{array} \begin{array}{c} \begin{array}{cccc} q^j_1 & q^j_2 & \cdots & q^j_{n(t_j)} \end{array} \\ \left[\begin{array}{c} \\ \\ (a^{ij}_{m(t_i)n(t_j)}) \\ \\ \end{array} \right] \end{array}$$

$1 \leq i \leq m$ and $1 \leq j \leq n$. S(m) is called the super dynamical FRM or a super dynamical system.

This matrix $A^{ij}_{m(t_i)n(t_j)}$ corresponds to the FRM matrix of the $(ij)^{th}$ expert with domain space attributes P^i_1, P^i_2, ..., $P^i_{m(t_i)}$ and range space attributes q^j_1, q^j_2, ..., $q^j_{n(t_j)}$, $1 \leq i \leq m$ and $1 \leq j \leq n$.

Super matrices are used as the dynamical system of SFRM models.

Next we proceed onto show how super row vectors (super row matrices) are used in the construction of SDBAM models. We recall the definition of SDBAM model.

DEFINITION 4.5: *Suppose a set of n experts choose to work with a problem using a BAM model in which they all agree upon the same number of attributes from the space F_x which will form the rows of the dynamical system formed by this multi expert BAM. Now n distinct sets of attributes are given from the space F_y which forms a super row vector and they form the columns of the BAM model.*

Suppose all the n experts agree to work with the same set of t-attributes say (x_1 x_2 ... x_t) which forms the rows of the synaptic connection matrix M. Suppose the first expert works with the p_1 set of attributes given by ($y^1_1\, y^1_2 \ldots y^1_{p_1}$), the second expert with

p_2 set of attributes given by $(y_1^1\ y_2^1 \ldots y_{p_1}^1)$ *and so on. Let the* i^{th} *expert with* p_i *set of attributes given by* $(y_1^i\ y_2^i \ldots y_{p_i}^i)$ *for i = 1, 2, ..., n. Thus the new BAM model will have its elements from* F_y *where any element in* F_y *will be a super row vector, T =* $(y_1^1\ y_2^1 \ldots y_{p_1}^1 \mid y_1^2\ y_2^2 \ldots y_{p_2}^2 \mid \ldots \mid y_1^n\ y_2^n \ldots y_{p_n}^n)$*. Now the synaptic projection matrix associated with this new BAM model is a special row supervector* M_r *given by*

$$M_r = \begin{array}{c} \\ x_1 \\ x_2 \\ \vdots \\ x_t \end{array} \begin{array}{c} \begin{array}{cccc} y_1^1\ y_2^1 \ldots y_{p_1}^1 & y_1^2\ y_2^2 \ldots y_{p_2}^2 & \ldots & y_1^n\ y_2^n \ldots y_{p_n}^n \end{array} \\ \left[\begin{array}{c|c|c|c} & & & \\ & & & \\ & & & \\ & & & \end{array}\right] \end{array}$$

Here the elements /attributes from F_x *is a simple row rector where as the elements from* F_y *is a super row vector.*

We call this model to be a multi expert Special Domain Supervector BAM (SDBAM) model and the associated matrix is a special row vector matrix denoted by M_r*. Let* $X = (x_1\ x_2\ \ldots\ x_t) \in F_x$*.* $Y = [y_1^1\ y_2^1 \ldots y_{p_1}^1 \mid y_1^2\ y_2^2 \ldots y_{p_2}^2 \mid \ldots \mid y_1^n\ y_2^n \ldots y_{p_n}^n] \in F_y$*. If* $X = (x_1\ x_2\ \ldots\ x_t) \in F_x$ *is the state vector given by the expert we find*

$$\begin{aligned} XM_r &= Y_1 \in F_y \\ YM_r^T &= X_1 \in F_x \ldots \end{aligned}$$

and so on. This procedure is continued until an equilibrium is arrived. Similarly if the expert chooses to work with Y = $[y_1^1\ y_2^1 \ldots y_{p_1}^1 \mid y_1^2\ y_2^2 \ldots y_{p_2}^2 \mid \ldots \mid y_1^n\ y_2^n \ldots y_{p_n}^n] \in F_y$ *then we find the resultant by finding*

$YM_r^T \hookrightarrow X$, then find XM_r and proceed on till the system arrives at an equilibrium state. This model will serve the purpose when row vectors from F_x are a simple row vectors and row vectors from F_y are super row vectors.

In the following we describe how the super column vectors (super column matrices) we used in the construction of synaptic matrix of the SRBAM models.

DEFINITION 4.6: *Suppose we have a problem in which all m experts want to work using a BAM model. If they agree to work having the simple vectors from F_y i.e., for the columns of the synaptic connection matrix i.e. there is no perpendicular partition of their related models matrix.*

The rows are partitioned horizontally in this synaptic connection matrix i.e., the m experts have distinct sets of attributes taken from the space F_x i.e. elements of F_x are super row vectors. The resulting synaptic connection matrix M_c is a special super column matrix.

Let the 1st expert have the set of row attributes to be $(x_1^1\ x_2^1 \ldots x_{q_1}^1)$, the 2nd expert have the set of row attributes given by $(x_1^2\ x_2^2 \ldots x_{q_2}^2)$ and so on. Let the ith expert have the related row attributes as $(x_1^i\ x_2^i \ldots x_{q_i}^i)$; $i = 1, 2, \ldots, m$.

Let the column vector given by all them is $[y_1 \ldots y_n]$. The related super synaptic connection matrix

$$M_C = \begin{array}{c} \\ x_1^1 \\ x_2^1 \\ \vdots \\ x_{q_1}^1 \\ x_1^2 \\ x_2^2 \\ \vdots \\ x_{q_2}^2 \\ \vdots \\ x_1^m \\ x_2^m \\ \vdots \\ x_{q_m}^m \end{array} \begin{array}{c} y_1 \quad y_2 \quad \ldots \quad y_n \\ \left[\begin{array}{c} \\ \\ \\ \qquad\qquad\qquad \\ \hline \\ \\ \\ \\ \hline \\ \hline \\ \\ \\ \\ \end{array} \right] \end{array}.$$

M_c is a special super column vector / matrix.

Suppose an expert wishes to work with a super row vector X from F_x then $X = [\, x_1^1 \; x_2^1 \ldots x_{q_1}^1 \mid x_1^2 \; x_2^2 \ldots x_{q_2}^2 \mid \ldots \mid x_1^m \; x_2^m \ldots x_{q_m}^m \,]$ we find $X \circ M_c \hookrightarrow Y \in F_y$, $YM_c^T = X_1 \in F_x$, we repeat the same procedure till the system attains its equilibrium i.e., a fixed point or a limit cycle.

This model which performs using the dynamical system M_c is defined as the Special Super Range BAM (SRBAM) model.

We now proceed onto describe the SDSBAM model which makes use of the super diagonal matrices. We describe mainly these models to show how these models makes use of the super matrices. This SDSBAM model synaptic connection matrix is a diagonal super matrix.

DEFINITION 4.7: *Suppose we have n experts to work on a specific problem and each expert wishes to work with a set of row and column attributes distinct from others using a BAM model. Then how to obtain a suitable integrated dynamical system using them. Let the first expert work with* $(x_1^1\ x_2^1 \ldots x_{n_1}^1)$ *attributes along the row of the related synaptic connection matrix of the related BAM and* $(y_1^1\ y_2^1 \ldots y_{p_1}^1)$ *the attributes related to the column, let the second expert give the row attributes of the synaptic connection matrix of the BAM to be* $(x_1^2\ x_2^2 \ldots x_{n_2}^2)$ *and that of the column be* $(y_1^2\ y_2^2 \ldots y_{p_2}^2)$ *and so on. Let the* i^{th} *expert give the row attributes of the synaptic connection matrix of the BAM to be* $(x_1^i\ x_2^i \ldots x_{n_i}^i)$ *and that of the column to be* $(y_1^i\ y_2^i \ldots y_{p_i}^i)$ *for i = 1, 2, …, n, the supermatrix described by*

$$
\begin{array}{c}
\\
M_D =
\end{array}
\begin{array}{c}
x_1^1 \\ \vdots \\ x_{n_1}^1 \\ x_1^2 \\ \vdots \\ x_{n_2}^2 \\ \vdots \\ x_1^n \\ \vdots \\ x_{n_n}^n
\end{array}
\begin{array}{c}
\begin{array}{cccc}
y_1^1\, y_2^1 \ldots y_{p_1}^1 & y_1^2\, y_2^2 \ldots y_{p_2}^2 & \ldots & y_1^n\, y_2^n \ldots y_{p_n}^n
\end{array} \\
\left[\begin{array}{c|c|c|c}
A_1^1 & (0) & & (0) \\ \hline
(0) & A_2^2 & & (0) \\ \hline
(0) & (0) & & (0) \\ \hline
(0) & (0) & & A_n^n
\end{array}\right]
\end{array}
$$

where A_i^i *is the synaptic connection matrix using the BAM model of the* i^{th} *expert, i = 1, 2, …, n where*

$$A^i{}_1 = \begin{array}{c} \\ x_1^i \\ x_2^i \\ \vdots \\ x_{n_i}^i \end{array} \begin{array}{c} \begin{array}{cccc} y_1^i & y_2^i & \cdots & y_{p_i}^i \end{array} \\ \left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array} \right] \end{array}$$

(0) denotes the zero matrix.

Thus this model has only non zero BAM synaptic connection matrices along the main diagonal described by M_D*. The rest are zero.*

The dynamical system associated with this matrix M_D *is defined to be the Special Diagonal Super BAM (SDSBAM) model.*

Yet another application (or use) of super matrices is in the super BAM models which is described below.

DEFINITION 4.8: *Suppose we have mn number of experts who are interested in working with a specific problem using a BAM model; a multi expert model which will work as a single dynamical system is given by the Super BAM (SBAM) model. Here a few experts have both the row and columns of the synaptic connection matrix of the BAM to be distinct. Some accept for same row attributes or vectors of the synaptic connection matrix but with different column attributes. Some accept for same column attributes of the synaptic connection matrix of the BAM model but with different row attributes to find the related supermatrix associated with the super BAM model. The supermatrix related with this new model will be denoted by* M_s *which is described in the following:*

$$
\begin{array}{c|c|c|c|c|}
 & y_1^1\, y_2^1 \ldots y_{q_1}^1 & y_1^2\, y_2^2 \ldots y_{q_2}^2 & \cdots & y_1^n\, y_2^n \ldots y_{q_n}^n \\
\hline
\begin{matrix} x_1^1 \\ \vdots \\ x_{p_1}^1 \end{matrix} & A_1^1 & A_2^1 & & A_n^1 \\
\hline
\begin{matrix} x_1^2 \\ \vdots \\ x_{p_2}^2 \end{matrix} & A_1^2 & A_2^2 & & A_n^2 \\
\hline
\vdots & & & & \\
\hline
\begin{matrix} x_1^m \\ \vdots \\ x_{p_m}^m \end{matrix} & A_1^m & A_2^m & & A_n^m
\end{array}
$$

where A_j^i *is the synaptic connection matrix of an expert who chooses to work with (* $x_1^i\ x_2^i \ldots x_{p_i}^i$ *) along the row of the BAM model and with (* $y_1^j\ y_2^j \ldots y_{q_j}^j$ *) along the column of the BAM model i.e.,*

$$
A_j^i = \begin{array}{c} \\ x_1^i \\ x_2^i \\ \vdots \\ x_{p_i}^i \end{array}
\begin{array}{c} y_1^j \quad y_2^j \quad \cdots \quad y_{q_j}^j \\ \left[\begin{array}{c} \\ \\ \\ \\ \end{array} \right] \end{array}
$$

$1 \leq i \leq m$ and $1 \leq j \leq n$. Thus for this model both the attributes from the spaces F_x and F_y are super row vectors given by

$$X = [\, x_1^1\ x_2^1 \ldots x_{p_1}^1 \mid x_1^2\ x_2^2 \ldots x_{p_2}^2 \mid \ldots \mid x_1^m\ x_2^m \ldots x_{p_m}^m \,]$$

in F_x and

$$Y = [\, y_1^1 \; y_2^1 \ldots y_{q_1}^1 \mid y_1^2 \; y_2^2 \ldots y_{q_2}^2 \mid \ldots \mid y_1^n \; y_2^n \ldots y_{q_n}^n \,]$$

from the space or the neuronal field F_y.

The supermatrix M_s is called the synaptic connection supermatrix associated with the multi expert super BAM model (SBAM model). Now having defined the multi expert super BAM model we proceed on to describe the functioning of the super dynamical system.

Let $X = [\, x_1^1 \; x_2^1 \ldots x_{p_1}^1 \mid x_1^2 \; x_2^2 \ldots x_{p_2}^2 \mid \ldots \mid x_1^m \; x_2^m \ldots x_{p_m}^m \,] \in F_x$ *be the super row vector given by the expert, its effect on the multi super dynamical system M_s.*

$$X \, o \, M_s \hookrightarrow Y$$

$$= [\, y_1^1 \; y_2^1 \ldots y_{q_1}^1 \mid y_1^2 \; y_2^2 \ldots y_{q_2}^2 \mid \ldots \mid y_1^n \; y_2^n \ldots y_{q_n}^n \,] \in F_y$$

$$Y \, o \, M_s^T \hookrightarrow X_1 \in F_x.$$

$$X_1 \, o \, M_s \hookrightarrow Y_1 \in F_y;$$

and so on and this procedure is repeated until the system attains a equilibrium.

Finally we just indicate how the super matrices are used in super fuzzy associative memories. For more about this refer [48].

A fuzzy set is a map μ : X → [0, 1] where X is any set called the domain and [0, 1] the range i.e., μ is thought of as a membership function i.e., to every element x ∈ X, μ assigns a membership value in the interval [0, 1]. But very few try to visualize the geometry of fuzzy sets. It is not only of interest but is meaningful to see the geometry of fuzzy sets when we discuss fuzziness. Till date researchers over looked such visualization

[24], instead they have interpreted fuzzy sets as generalized indicator or membership functions; i.e., mappings μ from domain X to range [0, 1]. But functions are hard to visualize. Fuzzy theorist often picture membership functions as two-dimensional graphs with the domain X represented as a one-dimensional axis.

The geometry of fuzzy sets involves both domain X = $(x_1, \ldots, x_n)$ and the range [0, 1] of mappings $\mu : X \to [0, 1]$. The geometry of fuzzy sets aids us when we describe fuzziness, define fuzzy concepts and prove fuzzy theorems. Visualizing this geometry may by itself provide the most powerful argument for fuzziness.

An odd question reveals the geometry of fuzzy sets. What does the fuzzy power set $F(2^X)$, the set of all fuzzy subsets of X, look like? It looks like a cube, What does a fuzzy set look like? A fuzzy subsets equals the unit hyper cube $I^n = [0, 1]^n$. The fuzzy set is a point in the cube I^n. Vertices of the cube I^n define a non-fuzzy set. Now with in the unit hyper cube $I^n = [0, 1]^n$ we are interested in a distance between points, which led to measures of size and fuzziness of a fuzzy set and more fundamentally to a measure. Thus within cube theory directly extends to the continuous case when the space X is a subset of R^n.

The next step is to consider mappings between fuzzy cubes. This level of abstraction provides a surprising and fruitful alternative to the prepositional and predicate calculus reasoning techniques used in artificial intelligence (AI) expert systems. It allows us to reason with sets instead of propositions. The fuzzy set framework is numerical and multidimensional. The AI framework is symbolic and is one dimensional with usually only bivalent expert rules or propositions allowed. Both frameworks can encode structured knowledge in linguistic form. But the fuzzy approach translates the structured knowledge into a flexible numerical framework and processes it in a manner that resembles neural network processing. The numerical framework also allows us to adaptively infer and modify fuzzy systems perhaps with neural or statistical techniques directly from problem domain sample data.

Between cube theory is fuzzy-systems theory. A fuzzy set defines a point in a cube. A fuzzy system defines a mapping between cubes. A fuzzy system S maps fuzzy sets to fuzzy sets. Thus a fuzzy system S is a transformation S: $I^n \to I^P$. The n-dimensional unit hyper cube I^n houses all the fuzzy subsets of the domain space or input universe of discourse X = $\{x_1, \ldots, x_n\}$. I^p houses all the fuzzy subsets of the range space or output universe of discourse, Y = $\{y_1, \ldots, y_p\}$. X and Y can also denote subsets of R^n and R^p. Then the fuzzy power sets F (2^X) and F (2^Y) replace I^n and I^p.

In general a fuzzy system S maps families of fuzzy sets to families of fuzzy sets thus S: $I^{n_1} \times \ldots \times I^{n_r} \to I^{p_1} \times \ldots \times I^{p_s}$ Here too we can extend the definition of a fuzzy system to allow arbitrary products or arbitrary mathematical spaces to serve as the domain or range spaces of the fuzzy sets. We shall focus on fuzzy systems S: $I^n \to I^P$ that map balls of fuzzy sets in I^n to balls of fuzzy set in I^p. These continuous fuzzy systems behave as associative memories. The map close inputs to close outputs. We shall refer to them as Fuzzy Associative Maps or FAMs.

The simplest FAM encodes the FAM rule or association (A_i, B_i), which associates the p-dimensional fuzzy set B_i with the n-dimensional fuzzy set A_i. These minimal FAMs essentially map one ball in I^n to one ball in I^p. They are comparable to simple neural networks. But we need not adaptively train the minimal FAMs. As discussed below, we can directly encode structured knowledge of the form, "If traffic is heavy in this direction then keep the stop light green longer" is a Hebbian-style FAM correlation matrix. In practice we sidestep this large numerical matrix with a virtual representation scheme. In the place of the matrix the user encodes the fuzzy set association (Heavy, longer) as a single linguistic entry in a FAM bank linguistic matrix. In general a FAM system F: $I^n \to I^b$ encodes the processes in parallel a FAM bank of m FAM rules (A_1, B_1), …, (A_m B_m). Each input A to the FAM system activates each stored

FAM rule to different degree. The minimal FAM that stores (A_i, B_i) maps input A to B_i' a partly activated version of B_i. The more A resembles A_i, the more B_i' resembles B_i. The corresponding output fuzzy set B combines these partially activated fuzzy sets $B_1^1, B_2^1, \ldots, B_m^1$. B equals a weighted average of the partially activated sets B = $w_1 B_1^1 + \ldots + w_m B_m^1$ where w_i reflects the credibility frequency or strength of fuzzy association (A_i, B_i). In practice we usually defuzzify the output waveform B to a single numerical value y_j in Y by computing the fuzzy centroid of B with respect to the output universe of discourse Y.

More generally a FAM system encodes a bank of compound FAM rules that associate multiple output or consequent fuzzy sets $B^1{}_i$, …, B_i^s with multiple input or antecedent fuzzy sets A_i^1, …, A_i^r. We can treat compound FAM rules as compound linguistic conditionals. This allows us to naturally and in many cases easily to obtain structural knowledge. We combine antecedent and consequent sets with logical conjunction, disjunction or negation. For instance, we could interpret the compound association (A^1, A^2, B), linguistically as the compound conditional "IF X^1 is A^1 AND X^2 is A^2, THEN Y is B" if the comma is the fuzzy association (A^1, A^2, B) denotes conjunction instead of say disjunction.

We specify in advance the numerical universe of discourse for fuzzy variables X^1, X^2 and Y. For each universe of discourse or fuzzy variable X, we specify an appropriate library of fuzzy set values A_1^r, …, A_k^2 Contiguous fuzzy sets in a library overlap. In principle a neural network can estimate these libraries of fuzzy sets. In practice this is usually unnecessary. The library sets represent a weighted though overlapping quantization of the input space X. They represent the fuzzy set values assumed by a fuzzy variable. A different library of fuzzy sets similarly quantizes the output space Y. Once we define the library of fuzzy sets we construct the FAM by choosing appropriate combinations of input and output fuzzy sets Adaptive techniques can make, assist or modify these choices.

An Adaptive FAM (AFAM) is a time varying FAM system. System parameters gradually change as the FAM system samples and processes data. Here we discuss how natural network algorithms can adaptively infer FAM rules from training data. In principle, learning can modify other FAM system components, such as the libraries of fuzzy sets or the FAM-rule weights w_i.

We now define the notion of super fuzzy associative memories of different types.

DEFINITION 4.9: *We have a problem P on which n experts wishes to work using a FAM model which can work as a single unit multi expert system. Suppose all the n-experts agree to work with the same set of attributes from the domain space and they want to work with different and distinct sets of attributes from the range space. Suppose all the n experts wish to work with the domain attributes (x_1 x_2 ... x_t) from the cube* $I^t = \underbrace{[0, 1] \times \ldots \times [0, 1]}_{t-times}$. *Let the first expert work with the range attributes* $(y_1^1\ y_2^1\ \ldots\ y_{p_1}^1)$ *and the second expert works with the range attributes* $(y_1^2\ y_2^2\ \ldots\ y_{p_2}^2)$ *and so on. Thus the* i^{th} *expert works with the range attributes* $(y_1^i\ y_2^i\ \ldots\ y_{p_i}^i)$*, i = 1, 2, ..., n.*

Thus the range attributes

$$Y = (y_1^1\ y_2^1\ \ldots\ y_{p_1}^1 \mid y_1^2\ y_2^2\ \ldots\ y_{p_2}^2 \mid \ldots \mid y_1^n\ y_2^n\ \ldots\ y_{p_n}^n)$$

are taken from the cube $I^{p_1+p_2+\ldots+p_n} = \underbrace{[0, 1] \times [0, 1] \times \ldots \times [0, 1]}_{p_1+p_2+\ldots+p_n\ times}$.

This we see the range attributes are super row fuzzy vectors.

Now the matrix which can serve as the dynamical systems for this FAM model is given by F_R.

$$\begin{array}{c} \\ x_1 \\ \vdots \\ x_t \end{array} \overset{\begin{array}{cccc} y_1^1\, y_2^1 \ldots y_{p_1}^1 & y_1^2\, y_2^2 \ldots y_{p_2}^2 & \ldots & y_1^n\, y_2^n \ldots y_{p_n}^n \end{array}}{\left[\begin{array}{c|c|c|c} A_1^1 & A_2^1 & & A_n^1 \end{array}\right]}$$

Clearly F_R is a special super row fuzzy vector. Thus F: I^t Ʊ $I^{p_1+p_2+\ldots+p_n}$. Suppose using an experts opinion we have a fit vector, A = (a_1, a_2, ..., a_t) ; a_i ∈{0,1}, then A o F_R = max min (a_i, f_{ij}); a_i ∈ A and f_{ij} ∈ F_R. Let A o F_R = B = (b_j), then F_R o B = max min (f_{ij}, b^i_j) and so on, till we arrive at a fixed point or a limit cycle. The resultant fit vectors give the solution. This F_R gives the dynamical system of the new model which we call as the Fuzzy Special Super Row vector FAM model (SRFAM model).

DEFINITION 4.10: *Suppose we have n experts working on a problem and they agree upon to work with the same range attributes and wish to work with distinct domain attributes using a FAM model. We built a new FAM model called the special super column fuzzy vector FRM model (SCFAM) and its related matrix is denoted by F_c. The fit vectors of the domain space are simple fit vectors where as the fit vectors of the range space are super row fit vectors.*

Now we describe the special column fuzzy vector FAM, F_c by the following matrix. The column attributes of the super fuzzy dynamical system F_c are given by

$$(y_1\, y_2\, \ldots\, y_s) \in I^s = \underbrace{[0, 1]\times\ldots\times[0, 1]}_{s-times}.$$

The row attributes of the first expert is given by $(x_1^1, x_2^1, \ldots, x_{p_1}^1)$, the row attributes of the second expert is given by $(x_1^2, x_2^2, \ldots, x_{p_2}^2)$. Thus the row attributes of the i^{th} expert is given by $(x_1^i\; x_2^i\; \ldots\; x_{p_i}^i)$, i = 1, 2, ..., n.

We have

$$F_c = \begin{array}{c} \\ x_1^1 \\ \vdots \\ x_{p_1}^1 \\ x_1^2 \\ \vdots \\ x_{p_2}^2 \\ \vdots \\ x_1^n \\ \vdots \\ x_{p_n}^n \end{array} \begin{array}{c} y_1 \quad y_2 \quad \cdots \quad y_s \\ \left[\begin{array}{c} A_1 \\ \hline A_2 \\ \hline \vdots \\ \hline A_n \end{array}\right] \end{array}$$

to be a special super column fuzzy vector / matrix, where

$$A_i = \begin{array}{c} \\ x_1^i \\ x_2^i \\ \vdots \\ x_{p_i}^i \end{array} \begin{array}{c} y_1 \quad y_2 \quad \cdots \quad y_s \\ \left[\begin{array}{c} \\ \\ \\ \\ \end{array}\right] \end{array}$$

i = 1, 2, ..., n is a fuzzy $p_i \times s$ matrix. Suppose the expert wishes to work with a fit vector (say) [$x_1^1\ x_2^1 \ldots x_{p_1}^1 \mid x_1^2\ x_2^2 \ldots x_{p_2}^2 \mid \ldots \mid x_1^n\ x_2^n \ldots x_{p_n}^n$]. Then X o F_c= B where B is a simple row vector we find F_c *o* $B = \underbrace{[0,\ 1] \times \ldots \times [0,\ 1]}_{p_1+p_2+\ldots+p_n \text{ times}} = I_{p_1+\ldots+p_n}$ *; we proceed on to work till we arrive at an equilibrium state of the system.*

Next we proceed on to define FAM when n expert give opinion having distinct set of domain attributes and distinct set of range attributes.

DEFINITION 4.11: *Let n experts give opinion on a problem P and wish to use a FAM model, to put this data as an integrated multi expert system.*

Let the first expert give his/her attributes along the column as $(y_1^1\ y_2^1 \ldots y_{q_1}^1)$ *and those attributes along the row as* $(x_1^1\ x_2^1 \ldots x_{p_1}^1)$.

Let $(y_1^2\ y_2^2 \ldots y_{q_2}^2)$ *and* $(x_1^2\ x_2^2 \ldots x_{p_2}^2)$ *be the column and row attributes respectively given by the second expert and so on.*

Thus any i^{th} *expert gives the row and column attributes as* $(x_1^i\ x_2^i \ldots x_{p_i}^i)$ *and* $(y_1^i\ y_2^i \ldots y_{q_i}^i)$ *respectively, i =1, 2, 3, ..., n.*

So for any i^{th} *expert the associated matrix of the FAM would be denoted by* A^i *where*

$$A^i = \begin{array}{c} \\ x_1^i \\ x_2^i \\ \vdots \\ x_{p_i}^i \end{array} \begin{array}{c} y_1^i \quad y_2^i \quad \cdots \quad y_{q_i}^i \\ \left[\begin{array}{ccc} & & \phantom{y_{q_i}^i} \\ & & \\ & & \\ & & \end{array} \right] \end{array}$$

Now form the multi expert FAM model using these n FAM matrices A^1, A^2, ..., A^n *and get the multi expert system which is denoted by*

$$F_D = \begin{array}{c} \\ x_1^1 \\ \vdots \\ x_{p_1}^1 \\ x_1^2 \\ \vdots \\ x_{p_2}^2 \\ \vdots \\ x_1^n \\ \vdots \\ x_{p_n}^n \end{array} \begin{array}{c} \begin{array}{cccc} y_1^1\, y_2^1 \ldots y_{q_1}^1 & y_1^2\, y_2^2 \ldots y_{q_2}^2 & \ldots & y_1^n\, y_2^n \ldots y_{q_n}^n \end{array} \\ \left[\begin{array}{c|c|c|c} A^1 & (0) & & (0) \\ \hline (0) & A^2 & & (0) \\ \hline (0) & (0) & & (0) \\ \hline (0) & (0) & & A^n \end{array}\right] \end{array}.$$

This fuzzy supermatrix F_D will be known as the diagonal fuzzy supermatrix of the FAM and the multi expert system which makes use of this diagonal fuzzy supermatrix F_D will be known as the Fuzzy Super Diagonal FAM (SDFAM) model. Now the related fit fuzzy supervectors of this model F_x and F_y are fuzzy super row vectors given by $X = (x_1^1\; x_2^1 \ldots x_{p_1}^1 \mid x_1^2\; x_2^2 \ldots x_{p_2}^2 \mid \ldots \mid x_1^n\; x_2^n \ldots x_{p_n}^n) \in F_x$ and $Y = (y_1^1\; y_2^1 \ldots y_{q_1}^1 \mid y_1^2\; y_2^2 \ldots y_{q_2}^2 \mid \ldots \mid y_1^n\; y_2^n \ldots y_{q_n}^n) \in F_y$.

Now this new FAM model functions in the following way.

Suppose the expert wishes to work with the fuzzy super state fit vector $X = (x_1^1\; x_2^1 \ldots x_{p_1}^1 \mid x_1^2\; x_2^2 \ldots x_{p_2}^2 \mid \ldots \mid x_1^n\; x_2^n \ldots x_{p_n}^n)$ then $Y = (y_1^1\; y_2^1 \ldots y_{q_1}^1 \mid y_1^2\; y_2^2 \ldots y_{q_2}^2 \mid \ldots \mid y_1^n\; y_2^n \ldots y_{q_n}^n) \in F_y$.

Now F_D o $Y = X_1 \in F_x$ and X_i o $F_D = Y_1 \in F_y$ and so on.

This procedure is repeated until the system equilibrium is reached.

DEFINITION 4.12: *Let us suppose we have a problem for which mn experts want to give their opinion. Here some experts give distinct opinion both for the row attributes and column attributes. Some experts concur on the row attributes but give different column attributes and a few others have the same set of row attributes but have a different set of column attributes. All of them concur to work using the FAM model. To find a multi expert FAM model which can tackle and give solution to the problem simultaneously.*

To this end we make use of a super fuzzy matrix F_s which is described in the following. Let the mn experts give their domain and column attributes as follows.

The first expert works with the domain attributes as $(x_1^1\ x_2^1 \ldots x_{p_1}^1)$ and column attributes as $(y_1^1\ y_2^1 \ldots y_{q_1}^1)$. The second expert works with the same domain attributes viz $(x_1^1\ x_2^1 \ldots x_{p_1}^1)$ and but column attributes as $(y_1^2\ y_2^2 \ldots y_{q_2}^2)$. The i^{th} expert, $1 \le i \le n$ works with $(x_1^1\ x_2^1 \ldots x_{p_1}^1)$ as the domain attributes and $(y_1^i\ y_2^i \ldots y_{q_i}^i)$ as the column attributes.

The $(n + 1)^{th}$ experts works with the new set of domain attributes $(x_1^2\ x_2^2 \ldots x_{p_2}^2)$ but with the same set of column attributes viz. $(y_1^1\ y_2^1 \ldots y_{q_1}^1)$.

Now the $n + j^{th}$ expert works with using $(x_1^2\ x_2^2 \ldots x_{p_2}^2)$ as the domain attribute and $(y_1^i\ y_2^i \ldots y_{q_i}^i)$ as the column attribute $1 \le j \le n$. The $(2n + 1)^{th}$ expert works with $(x_1^3\ x_2^3 \ldots x_{p_3}^3)$ as the row attribute and $(y_1^1\ y_2^1 \ldots y_{q_1}^1)$ as the column attribute.

Thus any (2n + k)th *expert uses* $(x_1^2\ x_2^2 \ldots x_{p_2}^2)$ *to be the row attribute and* $(y_1^k\ y_2^k \ldots y_{q_k}^k)$ *to be the column attribute* $1 \leq k \leq n$. *Thus any* $(t_n + r)^{th}$ *expert works with* $(x_1^t\ x_2^t \ldots x_{p_t}^t)$ *as the row attribute* $(1 \leq t \leq m)$ *and* $(y_1^r\ y_2^r \ldots y_{q_r}^r)$ *as the column attribute* $1 \leq r \leq n$.

Now as $1 \leq t \leq m$ *and* $1 \leq r \leq n$ *we get the FAM matrices of all the mn experts which is given by the supermatrix* F_s.

$$
F_s = \begin{array}{c} \\ x_1^1 \\ \vdots \\ x_{p_1}^1 \\ x_1^2 \\ \vdots \\ x_{p_2}^2 \\ \vdots \\ x_1^m \\ \vdots \\ x_{p_m}^m \end{array}
\begin{array}{c} y_1^1\ y_2^1 \ldots y_{q_1}^1 \quad y_1^2\ y_2^2 \ldots y_{q_2}^2 \quad \ldots \quad y_1^n\ y_2^n \ldots y_{q_n}^n \\
\left[\begin{array}{c|c|c|c}
A_1^1 & A_2^1 & & A_n^1 \\ \hline
A_1^2 & A_2^2 & & A_n^2 \\ \hline
 & & & \\ \hline
A_1^m & A_2^m & & A_n^m
\end{array}\right]
\end{array}
$$

where A_j^i *is a fuzzy matrix associated with ij*th *expert*

$$A^{i}_{j} = \begin{array}{c} \\ x^i_1 \\ x^i_2 \\ \vdots \\ x^i_{p_i} \end{array} \overset{\begin{array}{cccc} y^j_1 & y^j_2 & \cdots & y^j_{q_j} \end{array}}{\left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array} \right]}$$

$1 \leq i \leq m$ and $1 \leq j \leq n$. This model is known as the multi expert fuzzy Super FAM (SFAM) model. The fit vectors associated with them are super row vectors from F_x and F_y.

The fit super row vector X from F_x is

$$X = [\, x^1_1 \; x^1_2 \; \ldots \; x^1_{p_1} \mid x^2_1 \; x^2_2 \; \ldots \; x^2_{p_2} \mid \ldots \mid x^m_1 \; x^m_2 \; \ldots \; x^m_{p_m} \,]$$

and the

$$X \in F_x = I^{p_1+p_2+\ldots+p_m} = \underbrace{[0,\, 1] \times [0,\, 1] \times \ldots \times [0,\, 1]}_{p_1+p_2+\ldots+p_m \text{ times}}.$$

The fit super row vector Y from F_y is

$$Y = (\, y^1_1 \; y^1_2 \; \ldots \; y^1_{q_1} \mid y^2_1 \; y^2_2 \; \ldots \; y^2_{q_2} \mid \ldots \mid y^n_1 \; y^n_2 \; \ldots \; y^n_{q_n} \,);$$

$$Y \in F_y = \mathrm{I}^{q_1+q_2+\ldots+q_n} = \underbrace{[0,\, 1] \times [0,\, 1] \times \ldots \times [0,\, 1]}_{q_1+q_2+\ldots+q_m \text{ times}}.$$

Thus if

$$X = [\, x^1_1 \; x^1_2 \; \ldots \; x^1_{p_1} \mid x^2_1 \; x^2_2 \; \ldots \; x^2_{p_2} \mid \ldots \mid x^m_1 \; x^m_2 \; \ldots \; x^m_{p_m} \,]$$

is the fit vector given by an expert; its effect on F_s is given by $X \text{ o } F_s = Y \in F_y$; now $F_s \text{ o } Y = X^1 \in F_x$ then find the effect of X^1 on F_s ; $X^1 \text{ o } F_s = Y^1 \in F_y$ and so on.

We repeat this procedure until we arrive at a equilibrium state of the system.

For exact use of the super matrices please refer [17]. We have shown how the abstract model is associated with a super matrix.

We have used super matrices to construct super linear algebra [45]. We have developed super linear algebras using super matrices is an analogous way like linear algebra and derived several classical theorems. Here we recall a few of them for more about the innovative use of super matrices refer [45].

DEFINITION 4.13: *Let $V = (V_1| \ldots |V_n)$ be a super vector space and $T_s = (T_1 | \ldots | T_n)$ be a linear operator in V. If $W = (W_1 | \ldots | W_n)$ be a super subspace of V; we say that $W = (W_1 | \ldots | W_n)$ is super invariant under T if for each super vector $\alpha = (\alpha_1 | \ldots |\alpha_n)$ in $W = (W_1 | \ldots | W_n)$ the super vector $T_s(\alpha)$ is in $W = (W_1 | \ldots | W_n)$ i.e. if $T_s(W)$ is contained in W. When the super subspace $W = (W_1 | \ldots | W_n)$ is super invariant under the operator $T_s = (T_1 | \ldots | T_n)$ then T_s induces a linear operator $(T_s)_W$ on the super subspace $W = (W_1 | \ldots | W_n)$.*

The linear operator $(T_s)_W$ is defined by $(T_s)_W(\alpha) = T_s(\alpha)$ for α in $W = (W_1 | \ldots | W_n)$ but $(T_s)_W$ is a different object from $T_s = (T_1 | \ldots | T_n)$ since its domain is W not V.

When $V = (V_1 | \ldots | V_n)$ is finite $(n_1, \ldots, n_n)$ dimensional, the invariance of $W = (W_1 | \ldots | W_n)$ under $T_s = (T_1 | \ldots | T_n)$ has a simple super matrix interpretation and perhaps we should mention it at this point. Suppose we choose an ordered basis $B = (B_1 | \ldots | B_n) = (\alpha_1^1 \ldots \alpha_{n_1}^1 | \ldots | \alpha_1^n \ldots \alpha_{n_n}^n)$ for $V = (V_1 | \ldots | V_n)$ such that $B' = (\alpha_1^1 \ldots \alpha_{r_1}^1 | \ldots | \alpha_1^n \ldots \alpha_{r_n}^n)$ is an ordered basis for $W = (W_1 | \ldots | W_n)$; super dim $W = (r_1, \ldots, r_n)$. Let $A = [T_s]_B$ so that

$$T_s\alpha_j = \left[\sum_{i_1=1}^{n_1} A^1_{i_1j_1}\alpha^1_{i_1} \mid \ldots \mid \sum_{i_n=1}^{n_n} A^n_{i_nj_n}\alpha^n_{i_n}\right].$$

Since $W = (W_1 \mid \ldots \mid W_n)$ is super invariant under $T_s = (T_1 \mid \ldots \mid T_n)$ and the vector $T_s\alpha_j = (T_1\alpha^1_{j_1} \mid \ldots \mid T_n\alpha^n_{j_n})$ belongs to $W = (W_1 \mid \ldots \mid W_n)$ for $j_t \leq r_t$. This means that

$$T_s\alpha_j = \left[\sum_{i_1=1}^{r_1} A^1_{i_1j_1}\alpha^1_{i_1} \mid \ldots \mid \sum_{i_n=1}^{r_n} A^n_{i_nj_n}\alpha^n_{i_n}\right]$$

$j_t \leq r_t$; $t = 1, 2, \ldots, n$. In other words $A^t_{i_tj_t} = (A^1_{i_1j_1} \mid \ldots \mid A^n_{i_nj_n}) = (0 \mid \ldots \mid 0)$ if $j_t \leq r_t$ and $i_t > r_t$.

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & \ldots & 0 \\ \hline 0 & A_2 & 0 & 0 \\ \hline \vdots & 0 & & 0 \\ \hline 0 & 0 & 0 & A_n \end{array}\right)$$

$$= \left(\begin{array}{c|c|c|c} \begin{matrix} B_1 & C_1 \\ 0 & D_1 \end{matrix} & 0 & 0 & 0 \\ \hline 0 & \begin{matrix} B_2 & C_2 \\ 0 & D_2 \end{matrix} & & 0 \\ \hline 0 & & & 0 \\ \hline 0 & 0 & 0 & \begin{matrix} B_n & C_n \\ 0 & D_n \end{matrix} \end{array}\right)$$

where B_t is an $r_t \times r_t$ matrix, C_t is a $r_t \times (n_t - r_t)$ matrix and D_t is an $(n_t - r_t) \times (n_t - r_t)$ matrix $t = 1, 2, \ldots, n$.

In view of this we prove the following interesting lemma.

LEMMA 4.1: *Let $W = (W_1 \mid \ldots \mid W_n)$ be an invariant super subspace for $T_s = (T_1 \mid \ldots \mid T_n)$. The characteristic super polynomial for the restriction operator $(T_s)_W = ((T_1)_{W_1} \mid \ldots \mid (T_n)_{W_n})$ divides the characteristic super polynomial for T_s. The minimal super polynomial for $(T_s)_w = ((T_1)_{W_1} \mid \ldots \mid (T_n)_{w_n})$ divides the minimal super polynomial for T_s.*

Proof: We have $[T_s]_B = A$ where $B = \{B_1 \ldots B_n\}$ is a super basis for $V = (V_1 \mid \ldots \mid V_n)$; with $B_i = \left\{\alpha_1^i \ldots \alpha_{n_i}^i\right\}$ a basis for V_i, this is true for each i, i = 1, 2, …, n. A is a super diagonal square matrix of the form

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & \cdots & 0 \\ \hline 0 & A_2 & 0 & 0 \\ \hline \cdots & & & 0 \\ \hline 0 & 0 & 0 & A_n \end{array}\right)$$

where each

$$A_i = \begin{pmatrix} B_i & C_i \\ 0 & D_i \end{pmatrix}$$

for i = 1, 2, …, n; i.e.

$$A = \left(\begin{array}{c|c|c|c} \begin{matrix} B_1 & C_1 \\ 0 & D_1 \end{matrix} & 0 & 0 & 0 \\ \hline 0 & \begin{matrix} B_2 & C_2 \\ 0 & D_2 \end{matrix} & 0 & 0 \\ \hline 0 & & & 0 \\ \hline 0 & 0 & 0 & \begin{matrix} B_n & C_n \\ 0 & D_n \end{matrix} \end{array}\right)$$

and $B = [(T_s)_w]_{B'}$ where B′ is a basis for the super vector subspace W = (W_1 | … | W_n) and B is a super diagonal square matrix; i.e.

$$B = \left(\begin{array}{c|c|c|c} B_1 & 0 & \cdots & 0 \\ \hline 0 & B_2 & & 0 \\ \hline \vdots & & & 0 \\ \hline 0 & 0 & 0 & B_n \end{array}\right).$$

Now using the block form of the super diagonal square matrix we have super det (xI – A) = super det (xI – B) × super det (xI – D)

i.e. (det ($xI_1 - A_1$) |…| det ($xI_n - A_n$))

$$= (\det(xI_1' - B_1)\det(xI_1'' - D_1) |\ldots| \det(xI_n' - B_n)\det(xI_n'' - D_n)).$$

This proves the restriction operator $(T_s)_W$ super divides the characteristic super polynomial for T_s. The minimal super polynomial for $(T_s)_W$ super divides the minimal super polynomial for T_s.

It is pertinent to observe that I_1', $I_1'', I_1, \ldots, I_n$ represents different identities i.e. of different order.

The K^{th} row of A has the form

$$A^K = \left(\begin{array}{cc|cc|c|cc} B_1^{K_1} & C_1^{K_1} & \multicolumn{2}{c|}{0} & & \multicolumn{2}{c}{0} \\ 0 & D_1^{K_1} & & & & & \\ \hline \multicolumn{2}{c|}{0} & B_2^{K_2} & C_2^{K_2} & & \multicolumn{2}{c}{0} \\ & & 0 & D_2^{K_2} & & & \\ \hline \multicolumn{2}{c|}{\vdots} & & & & & \\ \hline \multicolumn{2}{c|}{0} & \multicolumn{2}{c|}{0} & \cdots & B_n^{K_n} & C_n^{K_n} \\ & & & & & 0 & D_n^{K_n} \end{array}\right)$$

where $C_t^{K_t}$ is some $r_t \times (n_t - r_t)$ matrix; true for t = 1, 2, …, n. Thus any super polynomial which super annihilates A also super annihilates D.

Thus our claim made earlier that, the minimal super polynomial for B super divides the minimal super polynomial for A is established.

Thus we say a super subspace $W = (W_1 \mid \ldots \mid W_n)$ of the super vector space $V = (V_1 \mid \ldots \mid V_n)$ is super invariant under $T_s = (T_1 \mid \ldots \mid T_n)$ if $T_s(W) \subseteq W$ i.e. each $T_i(W_i) \subseteq W_i$; for i = 1, 2, …, n i.e., if $\alpha = (\alpha_1 \mid \ldots \mid \alpha_n) \in W$

then $T_s\alpha = (T_1\alpha_1 \mid \ldots \mid T_n\alpha_n)$ where

$$\alpha_1 = x_1^1\alpha_1^1 + \ldots + x_{r_1}^1\alpha_{r_1}^1 \ ;$$

$$\alpha_2 = x_1^2\alpha_1^2 + \ldots + x_{r_2}^2\alpha_{r_2}^1$$

and so on

$$\alpha_n = x_1^n\alpha_1^n + \ldots + x_{r_n}^n\alpha_{r_n}^n \ .$$

$$T_s\alpha = (t_1^1 x_1^1\alpha_1^1 + \ldots + t_{r_1}^1 x_{r_1}^1\alpha_{r_1}^1 \mid \ldots \mid t_1^n x_1^n\alpha_1^n + \ldots + t_{r_n}^n x_{r_n}^n\alpha_{r_n}^n) \ .$$

Now B described in the above theorem is a super diagonal matrix given by

$$B = \left(\begin{array}{cccc|cccc|c|cccc}
t_1^1 & 0 & \dots & 0 & & & & & & & & & \\
0 & t_2^1 & \dots & 0 & & 0 & & & \dots & & 0 & & \\
\vdots & \vdots & & \vdots & & & & & & & & & \\
0 & 0 & \dots & t_{r_1}^1 & & & & & & & & & \\
\hline
 & & & & t_1^2 & 0 & \dots & 0 & & & & & \\
 & 0 & & & 0 & t_2^2 & \dots & 0 & & & 0 & & \\
 & & & & \vdots & \vdots & & \vdots & & & & & \\
 & & & & 0 & 0 & \dots & t_{r_2}^2 & & & & & \\
\hline
 & & & & & & & & & & & & \\
\hline
 & & & & & & & & & t_1^n & 0 & \dots & 0 \\
 & 0 & & & & 0 & & & & 0 & t_2^n & \dots & 0 \\
 & & & & & & & & & \vdots & \vdots & & \vdots \\
 & & & & & & & & & 0 & 0 & \dots & t_{r_n}^n
\end{array}\right)$$

Thus the characteristic super polynomial of B i.e. $(T_s)_W)$ is

$$g = (g_1 \mid \dots \mid g_n) =$$
$$((x - c_1^1)^{e_1^1} \dots (x - c_{K_1}^1)^{e_{K_1}^1} \mid \dots \mid (x - c_1^n)^{e_1^n} \dots (x - c_{K_n}^n)^{e_{K_n}^n})$$

where $e_i^t = \dim W_i^t$ for i = 1, 2, …, K_t and t = 1, 2, …, n.

Now we proceed onto define T_s super conductor of any α into W = $(W_1 \mid \dots \mid W_n)$.

DEFINITION 4.14: *Let $V = (V_1 \mid \dots \mid V_n)$ be a super vector space over the field F. $W = (W_1 \mid \dots \mid W_n)$ be an invariant super subspace of V for the linear operator $T_s = (T_1 \mid \dots \mid T_n)$ of V. Let $\alpha = (\alpha_1 \mid \dots \mid \alpha_n)$ be a super vector in V. The T-super conductor of α into W is the set $S_{T_s}(\alpha;W) = (S_{T_1}(\alpha_1;W_1) \mid \dots \mid S_{T_n}(\alpha_n;W_n))$ which consist of all super polynomials $g = (g_1 \mid \dots \mid g_n)$ (over the scalar field F) such that $g(T_s)\alpha$ is in W, i.e. $(g_1(T_1)\alpha_1 \mid \dots \mid g_n(T_n)\alpha_n) \in W = (W_1 \mid \dots \mid W_n)$. i.e. $g_i(T_i)\alpha_i) \in W_i$ for every i. Or we can equivalently define the T_s – super conductor of α in W is* *a T_i conductor of α_i in W_i for every i = 1, 2, ..., n. Without loss in meaning we can for convenience drop T_s and write the super conductor of α into W as* $S(\alpha;W) = (S(\alpha_1;W_1)|\ldots|S(\alpha_n;W_n))$.

The collection of polynomials will be defined as super stuffer this implies that the super conductor, the simple super operator $g(T_s) = (g_1(T_1) \mid \ldots \mid g_n(T_n))$ leads the super vector α into W. In the special case W = (0 | ... | 0), the super conductor is called the T_s super annihilator of α.

The following important and interesting theorem is proved.

THEOREM 4.1: *Let $V = (V_1 \mid \ldots \mid V_n)$ be a finite dimensional super vector space over the field F and let T_s be a linear operator on V. Then T_s is super diagonalizable if and only if the minimal super polynomial for T_s has the form*

$$p = (p_1 \mid \ldots \mid p_n) = ((x-c_1^1)\ldots(x-c_{K_1}^1)|\ldots|(x-c_1^n)\ldots(x-c_{K_n}^n)]$$

where $(c_1^1\ldots c_{K_1}^1|\ldots|c_1^n\ldots c_{K_n}^n)$ *are such that each set* $c_1^t,\ldots,c_{K_t}^t$ *are distinct elements of F for t = 1, 2, ..., n.*

Proof: We have noted that if T_s is super diagonalizable, its minimal super polynomial is a product of distinct linear factors. To prove the converse let $W = (W_1 \mid \ldots \mid W_n)$ be the super subspace spanned by all of the characteristic super vectors of T_s and suppose $W = (W_1 \mid \ldots \mid W_n) \neq (V_1 \mid \ldots \mid V_n)$ i.e. each $W_i \neq V_i$. By the earlier results proved there is a super vector α not in $W = (W_1 \mid \ldots \mid W_n)$ and a characteristic super value $c_j = (c_{j_1}^1, \ldots c_{j_n}^n)$ of T_s such that the super vector $\beta = (T - c_j I)\alpha$ i.e. $(\beta_1 \mid\ldots|\beta_n) = ((T_1 - c_{j_1}^1 I_1)\alpha_1|\ldots|(T_n - c_{j_n}^n I_n)\alpha_n)$ lies in $W = (W_1 \mid \ldots \mid W_n)$. Since $(\beta_1 \mid \ldots \mid \beta_n)$ is in W,

$$\beta = (\beta_1^1 + \ldots + \beta_{K_1}^1 \mid \beta_1^2 + \ldots + \beta_{K_2}^2|\ldots|\beta_1^n + \ldots + \beta_{K_n}^n)$$

where $\beta_t = \beta_1^t + \ldots + \beta_{K_t}^t$ for t = 1, 2, …, n with $T_s\beta_i = c_i\beta_i$; $1 \leq i \leq K$ i.e. $(T_1\beta_{i_1}^1 |\ldots| T_n\beta_{i_n}^n) = (c_{i_1}^1 \beta_{i_1}^1 |\ldots| c_{i_n}^n \beta_{i_n}^n)$; $(1 \leq i_t \leq K_t)$ and therefore the super vector

$$\begin{aligned} h(T_s)\beta = (h_1(c_1^1)\beta_1^1 + \ldots + h_1(c_{K_1}^1)\beta_{K_1}^1 |\ldots| \\ h_n(c_1^n)\beta_1^n + \ldots + h_n(c_{K_n}^n)\beta_{K_n}^n) \\ = (h_1(T_1)\beta_1 |\ldots| h_n(T_n)\beta_n) \end{aligned}$$

is in W = $(W_1 | \ldots | W_n)$ for every super polynomial h = $(h_1 | \ldots | h_n)$.

Now $(x - c_j)$ q for some super polynomial q, where p = $(p_1 | \ldots | p_n)$ and q = $(q_1 | \ldots | q_n)$.

Thus p = $(x - c_j)$ q implies

$$\begin{aligned} p &= (p_1 | \ldots | p_n) \\ &= ((x - c_{j_1}^1)q_1 |\ldots| (x - c_{j_n}^n)q_n) \end{aligned}$$

i.e. $(q_1 - q_1(c_{j_1}^1) |\ldots| q_n - q_n(c_{j_n}^n)) = ((x - c_{j_1}^1)h_1 |\ldots| (x - c_{j_n}^n)h_n)$. We have

$$\begin{aligned} & q(T_s)\alpha - q(c_j)\alpha = (q_1(T_1)\alpha_1 - q_1(c_{j_1}^1)\alpha_1 |\ldots| \\ & q_n(T_n)\alpha_n - q_n(c_{j_n}^n)\alpha_n) \\ = \;& h(T_s)(T_s - c_j I)\alpha = h(T_s)\beta \\ = \;& (h_1(T_1)(T_1 - c_{j_1}^1 I_1)\alpha_1 |\ldots| h_n(T_n)(T_n - c_{j_n}^n I_n)\alpha_n) \\ & = (h_1(T_1)\beta_1 |\ldots| h_n(T_n)\beta_n). \end{aligned}$$

But $h(T_s)\beta$ is in W = $(W_1 | \ldots | W_n)$ and since

$$\begin{aligned} & 0 = p(T_s)\alpha = (p_1(T_1)\alpha_1 |\ldots| p_n(T_n)\alpha_n) \\ & = (T_s - c_j I)q(T_s)\alpha \\ & = ((T_1 - c_{j_1}^1 I_1)\, q_1(T_1)\alpha_1 |\ldots| (T_n - c_{j_n}^n I_n) q_n(T_n)\alpha_n); \end{aligned}$$

the vector q(T_s)α is in W. Therefore q(c_j)α is in W. Since α is not in W we have $q(c_j) = (q_1(c^1_{j_1})|\ldots|q_n(c^n_{j_n})) = (0|\ldots|0)$. Thus contradicts the fact that p = (p_1 | … | p_n) has distinct roots.

If T_s is represented by a super diagonal square matrix A in some super basis and we wish to know if T_s is super diagonalizable. We compute the characteristic super polynomial f = (f_1 | … | f_n). If we can factor

$$\begin{aligned} f &= (f_1 \mid \ldots \mid f_n) \\ &= ((x-c^1_1)^{d^1_1}\ldots(x-c^1_{K_1})^{d^1_{K_1}}|\ldots|(x-c^n_1)^{d^n_1}\ldots(x-c^n_{K_n})^{d^n_{K_n}}) \end{aligned}$$

we have two different methods for determining whether or not T is super diagonalizable. One method is to see whether for each i = 1, 2, …, n we can find $d^t_{i_t}$ independent characteristic super vectors associated with the characteristic super values c_{i_t}. The other method is to check whether or not

$$(T_s - c_1 I)\ldots(T_s - c_k I) \text{ i.e. } ((T_1 - c^1_1 I_1)\ldots(T_1 - c^1_{K_1} I_1)$$
$$|\ldots|(T_n - c^n_1)I_n \ldots (T_n - c^n_{K_n} I_n))$$

is the super zero operator.

Several other interesting results in this direction can be derived. Now we proceed onto define the new notion of super independent subsuper spaces of a super vector space V.

DEFINITION 4.15: *Let V = (V_1 | … | V_n) be a super vector space over F. Let $W_1 = (W^1_1 |\ldots|W^n_1)$, $W_2 = (W^n_2|\ldots|W^n_2)\ldots$ $W_K = (W^1_K |\ldots|W^n_K)$ be K super subspaces of V. We say W_1, …, W_K are super independent if $\alpha_1 + \ldots + \alpha_K = 0$; $\alpha_i \in W_i$ implies each $\alpha_i = 0$.*

$$\alpha_i = (\alpha^i_1 \mid \ldots \mid \alpha^i_n) \in W_i = (W^1_i|\ldots|W^n_i);$$

true for i = 1, 2, …, K. If W_1 and W_2 are any two super vector subspaces of V = (V_1 | … | V_n), we say $W_1 = (W_1^1 \mid \ldots \mid W_1^n)$ *and* $W_2 = (W_2^1 \mid \ldots \mid W_2^n)$ *are super independent if and only if* $W_1 \cap W_2 = (W_1^1 \cap W_2^1 \mid \ldots \mid W_1^n \cap W_2^n)$ *= (0 | 0 | … |0). If W_1, W_2, …, W_K are K super subspaces of V we say W_1, W_2, …, W_K are independent if $W_1 \cap W_2 \cap \ldots \cap W_K$ =* $(W_1^1 \cap W_2^1 \cap \ldots W_K^1 \mid \ldots \mid W_1^n \cap W_2^n \cap \ldots \cap W_K^n)$ *= (0 | … | 0) . The importance of super independence in super subspaces is mentioned below. Let*

$$\begin{aligned} W' &= W'_1 + \ldots + W'_k \\ &= (W_1^1 + \ldots + W_K^1 \mid \ldots \mid W_1^n + \ldots + W_K^n) \\ &= (W'_1 \mid \ldots \mid W'_n) \end{aligned}$$

W'_i *is a subspace V_i and* $W'_i = W'_1 + \ldots + W'_K$ *true for i = 1, 2, …, n. Each super vector α in W can be expressed as a sum* $\alpha = (\alpha'_1 \mid \ldots \mid \alpha'_n) = ((\alpha_1^1 + \ldots + \alpha_K^1) \mid \ldots \mid \alpha_1^n + \ldots + \alpha_K^n))$ *i.e. each* $\alpha^t = \alpha_1^t + \ldots + \alpha_K^t$; $\alpha^t \in W_t$*. If W_1, W_2, …, W_K are super independent, then that expression for α is unique; for if*

$$\alpha = (\beta_1 + \ldots + \beta_K) = (\beta_1^1 + \ldots + \beta_K^1 \mid \ldots \mid \beta_1^n + \ldots + \beta_K^n)$$

$\beta_i \in W_i$; *i = 1, 2, …, K.* $\beta_i = \beta_1^i + \ldots + \beta_n^i$ *then*

$$\alpha - \alpha = (0 \mid \ldots \mid 0) = ((\alpha_1^1 - \beta_1^1) + \ldots + (\alpha_K^1 - \beta_K^1) \mid \ldots \mid (\alpha_1^n - \beta_1^n) + \ldots + (\beta_K^n - \alpha_K^n))$$

hence each $\alpha_i^t - \beta_i^t = 0$ *; 1 ≤ i ≤ K; t = 1, 2, …, n. Thus W_1, W_2, …, W_K are super independent so we can operate with super vectors in W as K-tuples* $((\alpha_1^1, \ldots, \alpha_K^1); \ldots, (\alpha_1^n, \ldots, \alpha_K^n); \alpha_i^t \in W_t$ *; 1 ≤ i ≤ K; t = 1, 2, …, n. in the same way we operate with R^K as K-tuples of real numbers.*

LEMMA 4.2: *Let V = (V_1 | … | V_n) be a finite (n_1, …, n_n) dimensional super vector space. Let W_1, …, W_K be super* *subspaces of V and let* $W = (W_1^1 + \ldots W_K^1 | \ldots | W_1^n + \ldots + W_K^n)$. *The following are equivalent*

(a) *$W_1, \ldots, W_K$ are super independent.*
(b) *For each j; $2 \le j \le K$, we have $W_j \cap (W_1 + \ldots + W_{j-1}) = \{(0 | \ldots | 0)\}$*
(c) *If B_i is a super basis of W_i, $1 \le i \le K$, then the sequence $B = (B_1 \ldots B_K)$ is a super basis for W.*

The proof is left as an exercise for the reader. In any or all of the conditions of the above stated lemma is true then the supersum $W = W_1 + \ldots + W_K = (W_1^1 + \ldots + W_K^1 | \ldots | W_1^n + \ldots + W_K^n)$ where $W_t = (W_t^1 | \ldots | W_t^n)$ is super direct or that W is a super direct sum of $W_1, \ldots, W_K$ i.e. $W = W_1 \oplus \ldots \oplus W_K$ i.e. $(W_1^1 \oplus \ldots \oplus W_K^1 | \ldots | W_1^n \oplus \ldots \oplus W_K^n)$. If each of the W_i is (1, …, 1) dimensional then $W = W_1 \oplus \ldots \oplus W_n = (W_1^1 \oplus \ldots \oplus W_n^1 | \ldots | W_1^n \oplus \ldots \oplus W_n^n)$.

DEFINITION 4.16: *Let $V = (V_1 | \ldots | V_n)$ be a super vector space over the field F; a super projection of V is a linear operator E_s on V such that $E_s^2 = E_s$ i.e. $E_s = (E_1 | \ldots | E_n)$ then $E_s^2 = (E_1^2 | \ldots | E_n^2) = (E_1 | \ldots | E_n)$ i.e. each E_i is a projection on V_i; i = 1, 2, …, n.. Suppose E_s is a projection on V and $R = (R_1 | \ldots | R_n)$ is the super range of E_s and $N = (N_1 | \ldots | N_n)$ the super null space or null super space of E_s. The super vector $\beta = (\beta_1 | \ldots | \beta_n)$ is in the super range $R = (R_1 | \ldots | R_n)$ if and only if $E_s\beta = \beta$ i.e. if and only if $(E_1\beta_1 | \ldots | E_n\beta_n) = (\beta_1 | \ldots | \beta_n)$ i.e. each $E_i\beta_i = \beta_i$ for i = 1, 2, …, n.. If $\beta = E_s\alpha$ i.e. $\beta = (\beta_1 | \ldots | \beta_n) = (E_1\alpha_1 | \ldots | E_n\alpha_n)$ where the super vector $\alpha = (\alpha_1 | \ldots | \alpha_n)$ then $E_s\beta = E_s^2\alpha = E_s\alpha = \beta$. Conversely if $\beta = (\beta_1 | \ldots | \beta_n) = E_s\beta = (E_1\beta_1 | \ldots | E_n\beta_n)$ then $\beta = (\beta_1 | \ldots | \beta_n)$ is in the super range of E_s. Thus $V = R \oplus N$ i.e. $V = (V_1 | \ldots | V_n) = (R_1 \oplus N_1 | \ldots | R_n \oplus N_n)$.*

Further the unique expression for $= (\alpha_1 | \ldots | \alpha_n)$ as a sum of super vectors in R and N is $\alpha = E_s\alpha + (\alpha - E_s\alpha)$ i.e. $\alpha_i =$

$E_i\alpha_i + (\alpha_i - E_i\alpha_i)$ for i = 1, 2, …, n. From what we have stated it easily follows that if R and N are super subspace of V such that $V = R \oplus N$ i.e. $V = (V_1 \mid \ldots \mid V_n) = (R_1 \oplus N_1 \mid \ldots \mid R_n \oplus N_n)$ then there is one and only one super projection operator E_s which has super range $R = (R_1 \mid \ldots \mid R_n)$ and null super space $N = (N_1 \mid \ldots \mid N_n)$. That operator is called the super projection on R along N.

Any super projection E_s is super diagonalizable. If $\{(\alpha_1^1 \ldots \alpha_{r_1}^1 \mid \ldots \mid \alpha_1^n \ldots \alpha_{r_n}^n)$ is a super basis for $R = (R_1 \mid \ldots \mid R_n)$ and $(\alpha_{r_1+1}^1 \ldots \alpha_{n_1}^1 \mid \ldots \mid \alpha_{r_n+1}^n \ldots \alpha_{n_n}^n)$ is a super basis for $N = (N_1 \mid \ldots \mid N_n)$ then the basis $B = (\alpha_1^1 \ldots \alpha_{n_1}^1) \mid \ldots \mid \alpha_1^n \ldots \alpha_{n_n}^n) = (B_1 \mid \ldots \mid B_n)$ super diagonalizes E_s.

$$(E_s)_B = \left(\begin{array}{cc|c|c|cc} I_1 & 0 & \multirow{2}{*}{0} & \multirow{2}{*}{\cdots} & \multicolumn{2}{c}{\multirow{2}{*}{0}} \\ 0 & 0 & & & & \\ \hline \multicolumn{2}{c|}{0} & \begin{array}{cc} I_2 & 0 \\ 0 & 0 \end{array} & \cdots & \multicolumn{2}{c}{0} \\ \hline \multicolumn{2}{c|}{0} & 0 & & \multicolumn{2}{c}{0} \\ \hline \multicolumn{2}{c|}{0} & 0 & & I_n & 0 \\ & & & & 0 & 0 \end{array}\right)$$

$$= ([E_1]| \ldots |[E_n]_{B_n})$$

where I_t is a $r_t \times r_t$ identity matrix; t = 1, 2, …, n. Thus super projections can be used to describe super direct sum decompositions of the super vector space $V = (V_1 \mid \ldots \mid V_n)$.

THEOREM 4.2: *Let F be the field of real numbers or the field of complex numbers. Let A be a super diagonal matrix of the form*

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_n \end{array}\right)$$

be a ($n_1 \times n_1$, …, $n_n \times n_n$) matrix over F. The super function g = (g_1 | … | g_n) defined by g(X, Y) = Y^*AX is a positive superform on the super space $(F^{n_1 \times 1} \mid \ldots \mid F^{n_n \times 1})$ if and only if there exists an invertible super diagonal matrix

$$P = \left(\begin{array}{c|c|c|c} P_1 & 0 & & 0 \\ \hline 0 & P_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & P_n \end{array}\right).$$

Each P_i is a $n_i \times n_i$ matrix i = 1, 2, …, n with entries from F such that $A = P^* P$; i.e.,

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_n \end{array}\right)$$

$$= \left(\begin{array}{c|c|c|c} P_1^*P_1 & 0 & \ldots & 0 \\ \hline 0 & P_2^*P_2 & \ldots & 0 \\ \hline & & \ldots & \\ \hline 0 & 0 & \ldots & P_n^*P_n \end{array}\right).$$

DEFINITION 4.17: *Let*

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_n \end{array}\right)$$

be a superdiagonal matrix with each A_i a $n_i \times n_i$ matrix over the field F; i =1, 2, ..., n. The principal super minor of A or super principal minors of A (both mean the same) are scalars

$$\Delta_k(A) = (\Delta_{k_1}(A_1) | \ \ldots \ | \Delta_{k_n}(A_n))$$

defined by

$$\Delta_k(A) = superdet \left\{ \left(\begin{array}{c|c|c} \begin{matrix} A^1_{11} \ldots A^1_{1k_1} \\ \vdots \qquad \vdots \\ A^1_{k_1 1} \ldots A^1_{k_1 k_1} \end{matrix} & 0 & 0 \\ \hline 0 & \begin{matrix} A^2_{11} \ldots A^2_{1k_2} \\ \vdots \qquad \vdots \\ A^2_{k_2 1} \ldots A^2_{k_2 k_2} \end{matrix} & 0 \\ \hline 0 & 0 & \begin{matrix} A^n_{11} \ldots A^2_{1k_n} \\ \vdots \qquad \vdots \\ A^n_{k_n 1} \ldots A^n_{k_n k_n} \end{matrix} \end{array}\right) \right\}$$

$$= \left(\det \begin{pmatrix} A^1_{11} & \ldots & A^1_{1k_1} \\ \vdots & & \vdots \\ A^1_{k_1 1} & \ldots & A^1_{k_1 k_1} \end{pmatrix}, \ \ldots \ , \det \begin{pmatrix} A^n_{11} & \ldots & A^n_{1k_n} \\ \vdots & & \vdots \\ A^n_{k_{n1}} & \ldots & A^n_{k_n k_n} \end{pmatrix} \right)$$

for $1 \le k_t \le n_t$ and $t = 1, 2, \ldots, n$.

Several other interesting properties can also be derived for these superdiagonal matrices.

We give the following interesting theorem and the proof is left for the reader.

THEOREM 4.3: *Let f = (f_1 | ... | f_n) be a superform on a finite (n_1, ..., n_n) dimensional supervector space V = (V_1 | ... | V_n) and let A be a super diagonal matrix of f in an ordered superbasis B = (B_1 | ... | B_n). Then f is a positive superform if and only if A = A* and the principal super minor of A are all positive.*
i.e.,

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_n \end{array}\right)$$

$$= \left(\begin{array}{c|c|c|c} A_1^* & 0 & & 0 \\ \hline 0 & A_2^* & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_n^* \end{array}\right).$$

***Note**:* The principal minor of (A_1 | … | A_n) is called as the principal superminors of A or with default of notation the principal minors of {A_1, …, A_n} is called the principal super minors of A.

T_s = (T_1 | … | T_n) a linear operator on a finite (n_1, …, n_n) dimensional super inner product space V = (V_1 | … | V_n) is said to be super non-negative if $T_s = T_s^*$

i.e.,

$$(T_1 \mid \ldots \mid T_n) = (T_1^* \mid \ldots \mid T_n^*)$$

i.e. $T_i = T_i^*$ for i = 1, 2, …, n and

$$(T_s \alpha \mid \alpha) = ((T_1 \alpha_1 \mid \alpha_1) \mid \ldots \mid (T_n \alpha_n \mid \alpha_n)) \geq (0 \mid \ldots \mid 0)$$

for all $\alpha = (\alpha_1 \mid \ldots \mid \alpha_n)$ in V.

A super positive linear operator is one such that $T_s = T_s^*$ and

$$(T \alpha \mid \alpha) = ((T_1 \alpha_1 \mid \alpha_1) \mid \ldots \mid (T_n \alpha_n \mid \alpha_n)) > (0 \mid \ldots \mid 0)$$

for all $\alpha = (\alpha_1 \mid \ldots \mid \alpha_n) \neq (0 \mid \ldots \mid 0)$.

Several properties enjoyed by positive operators and non negative operators will also be enjoyed by the super positive operators and super non negative operators on super vector spaces, with pertinent and appropriate modification. Throughout the related matrix for these super operators T_s will always be a super diagonal matrix A of the form

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_n & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_n \end{array}\right)$$

where each A_i is a $n_i \times n_i$ square matrix, $1 \leq i \leq n$, $A = A^*$ and the principal minors of each A_i are positive; $1 \leq i \leq n$.

Now we just mention one more property about the super forms.

THEOREM 4.4: *Let $f = (f_1 \mid \ldots \mid f_n)$ be a super form on a real or complex super vector space $V = (V_1 \mid \ldots \mid V_n)$ and $\left\{\alpha_1^1 \ldots \alpha_{r_1}^1 \mid \ldots \mid \alpha_1^n \ldots \alpha_{r_n}^n\right\}$ a super basis for the finite dimensional super subvector space $W = (W_1 \mid \ldots \mid W_n)$ of $V = (V_1 \mid \ldots \mid V_n)$. Let M be the super square diagonal matrix where each M_i in M; is a $r_i \times r_i$ super matrix with entries ($1 \leq i \leq n$). $M_{jk}^i = f_i(\alpha_k^i, \alpha_j^i)$, i.e.*

$$M = \left(\begin{array}{c|c|c|c} M_1 & 0 & & 0 \\ \hline 0 & M_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & M_n \end{array}\right)$$

$$= \left(\begin{array}{c|c|c|c} f^1(\alpha^1_{k_1}, \alpha^1_{j_1}) & 0 & & 0 \\ \hline 0 & f^2(\alpha^2_{k_2}, \alpha^2_{j_2}) & & 0 \\ \hline & & & \\ \hline 0 & 0 & & f^n(\alpha^n_{k_n}, \alpha^n_{j_n}) \end{array}\right)$$

and $W' = (W_1' | \ldots | W_n')$ *be the set of all super vectors* $\beta = (\beta_1 \mid \ldots \mid \beta_n)$ *of V and* $W \cap W' = (W_1 \cap W_1' \mid \ldots \mid W_n \cap W_n') = (0 \mid \ldots \mid 0)$ *if and only if*

$$M = \left(\begin{array}{c|c|c|c} M_1 & 0 & & 0 \\ \hline 0 & M_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & M_n \end{array}\right)$$

is invertible. When this is the case, $V = W + W'$ *i.e.* $V = (V_1 \mid \ldots \mid V_n) = (W_1 + W_1' \mid \ldots \mid W_n + W_n')$.

The proof can be obtained as a matter of routine.

The projection $E_s = (E_1 \mid \ldots \mid E_n)$ constructed in the proof may be characterized as follows:

$$E_s\beta = \alpha;$$
$$(E_1\beta_1 \mid \ldots \mid E_n\beta_n) = (\alpha_1 \mid \ldots \mid \alpha_n)$$

is in W and $\beta - \alpha$ belongs $W' = (W_1' \mid \ldots \mid W_n')$. Thus E_s is independent of the super basis of $W = (W_1 \mid \ldots \mid W_n)$ that was used in this construction. Hence we may refer to E_s as the super projection of V on W that is determined by the direct sum decomposition.

$$V = W \oplus W';$$
$$(V_1 \mid \ldots \mid V_n) = (W_1 \oplus W_1' \mid \ldots \mid W_n \oplus W_n').$$

Note that E_s is a super orthogonal projection if and only if $W' = W^{\perp} = (W_1^{\perp} \mid \ldots \mid W_n^{\perp})$. Now we proceed onto develop the analogous of spectral theorem which we call as super spectral theorem.

Finally these super matrices are used to develop the notion of Leontief economic super models.

Matrix theory has been very successful in describing the interrelations between prices, outputs and demands in an economic model. Here we just discuss some simple models based on the ideals of the Nobel-laureate Massily Leontief. Two types of models discussed are the closed or input-output model and the open or production model each of which assumes some economic parameter which describe the inter relations between the industries in the economy under considerations. Using matrix theory we evaluate certain parameters.

The basic equations of the input-output model are the following:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix} = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix}$$

each column sum of the coefficient matrix is one

i. $p_i \geq 0$, $i = 1, 2, \ldots, n$.
ii. $a_{ij} \geq 0$, $i, j = 1, 2, \ldots, n$.

$$\text{iii.} \quad a_{ij} + a_{2j} + \ldots + a_{nj} = 1$$

for j = 1, 2 , …, n.

$$p = \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{pmatrix}$$

are the price vector. $A = (a_{ij})$ is called the input-output matrix

$$Ap = p \text{ that is, } (I - A)\, p = 0.$$

Thus A is an exchange matrix, then Ap = p always has a nontrivial solution p whose entries are nonnegative. Let A be an exchange matrix such that for some positive integer m, all of the entries of A^m are positive. Then there is exactly only one linearly independent solution of (I – A) p = 0 and it may be chosen such that all of its entries are positive in Leontief open production model.

In contrast with the closed model in which the outputs of k industries are distributed only among themselves, the open model attempts to satisfy an outside demand for the outputs. Portions of these outputs may still be distributed among the industries themselves to keep them operating, but there is to be some excess some net production with which to satisfy the outside demand. In some closed model, the outputs of the industries were fixed and our objective was to determine the prices for these outputs so that the equilibrium condition that expenditures equal incomes was satisfied.

x_i = monetary value of the total output of the i^{th} industry.

d_i = monetary value of the output of the i^{th} industry needed to satisfy the outside demand.

σ_{ij} = monetary value of the output of the i^{th} industry needed by the j^{th} industry to produce one unit of monetary value of its own output.

With these qualities we define the production vector.

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{pmatrix}$$

the demand vector

$$d = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_k \end{pmatrix}$$

and the consumption matrix,

$$C = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1k} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2k} \\ \vdots & \vdots & & \vdots \\ \sigma_{k1} & \sigma_{k2} & \cdots & \sigma_{kk} \end{pmatrix}.$$

By their nature we have

$$x \geq 0, \ d \geq 0 \text{ and } C \geq 0.$$

From the definition of σ_{ij} and x_j it can be seen that the quantity

$$\sigma_{i1}\, x_1 + \sigma_{i2}\, x_2 + \ldots + \sigma_{ik}\, x_k$$

is the value of the output of the i^{th} industry needed by all k industries to produce a total output specified by the production vector x.

Since this quantity is simply the i^{th} entry of the column vector Cx, we can further say that the i^{th} entry of the column vector x – Cx is the value of the excess output of the i^{th} industry available to satisfy the outside demand. The value of the outside demand for the output of the i^{th} industry is the i^{th} entry of the demand vector d; consequently; we are led to the following equation:

$$x - Cx = d \text{ or}$$
$$(I - C)\, x = d$$

for the demand to be exactly met without any surpluses or shortages. Thus, given C and d, our objective is to find a production vector $x \geq 0$ which satisfies the equation $(I - C)x = d$.

A consumption matrix C is said to be productive if $(1 - C)^{-1}$ exists and $(1 - C)^{-1} \geq 0$.

A consumption matrix C is productive if and only if there is some production vector $x \geq 0$ such that $x > Cx$.

A consumption matrix is productive if each of its row sums is less than one. A consumption matrix is productive if each of its column sums is less than one.

Now we will formulate the Smarandache analogue for this, at the outset we will justify why we need an analogue for those two models.

Clearly, in the Leontief closed Input - Output model,

p_i = price charged by the i^{th} industry for its total output in reality need not be always a positive quantity for due to competition to capture the market the price may be fixed at a loss or the demand for that product might have fallen down so badly so that the industry may try to charge very less than its real value just to market it.

Similarly $a_{ij} \geq 0$ may not be always be true. Thus in the Smarandache Leontief closed (Input-Output) model (S-Leontief closed (Input-Output) model) we do not demand $p_i \geq 0$, p_i can be negative; also in the matrix $A = (a_{ij})$,

$$a_{1j} + a_{2j} + \ldots + a_{kj} \neq 1$$

so that we permit a_{ij}'s to be both positive and negative, the only adjustment will be we may not have $(I - A)\, p = 0$, to have only one linearly independent solution, we may have more than one and we will have to choose only the best solution.

As in this complicated real world problems we may not have in practicality such nice situation. So we work only for the best solution.

Here we introduce a input-output model which has some p number of input-output matrix each of same order say $n \times n$ functioning simultaneously.

We shall call such models as input – output super row matrix models and describe how it functions. Suppose we have p number of $n \times n$ input output matrix given by the super row matrix $A = [A_1 \mid \ldots \mid A_n]$ where each A_i is a $n \times n$ input output matrix which are distinct.

$$A = [A_1 \mid \ldots \mid A_n]$$

$$= \left(\begin{pmatrix} a_{11}^1 & \ldots & a_{1n}^1 \\ a_{21}^1 & \ldots & a_{2n}^1 \\ \vdots & & \vdots \\ a_{n1}^1 & \ldots & a_{nn}^1 \end{pmatrix} \Bigg| \ldots \Bigg| \begin{pmatrix} a_{11}^p & \ldots & a_{1n}^p \\ a_{21}^p & \ldots & a_{2n}^p \\ \vdots & & \vdots \\ a_{n1}^p & \ldots & a_{nn}^1 \end{pmatrix} \right)$$

where $a_{ij}^t + a_{2j}^t + \ldots + a_{nj}^t = 1$; $t = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, n$. Suppose

$$P = \begin{pmatrix} p_1^1 & p_1^2 & \cdots & p_1^P \\ \vdots & \vdots & & \vdots \\ p_n^1 & p_n^2 & \cdots & p_n^P \end{pmatrix} = [P_1 \mid \ldots \mid P_p]$$

be the super column price vector then

A * P = P, the (product) * is defined as A * P = P that is

$$[A_1P_1 \mid \ldots \mid A_pP_p] = [P_1 \mid \ldots \mid P_p]$$
$$A * P = P$$

that is

$(I – A) P = (0 \mid \ldots \mid 0)$
i.e., $((I – A_1) P_1 \mid \ldots \mid (I – A_p) P_p) = (0 \mid \ldots \mid 0)$.

Thus A is an super-row square exchange matrix, then AP = P always has a row column vector solution P whose entries are non negative.

Let $A = [A_1 \mid \ldots \mid A_n]$ be an exchange super row square matrix such that for some positive integer m all the entries of A^m i.e. entries of each A_t^m are positive for m; m = 1, 2, …, p. Then there is exactly only one linearly independent solution of

$(I – A) P = (0 \mid \ldots \mid 0)$
i.e., $((I – A_1) P_1 \mid \ldots \mid (I – A_p) P_p) = (0 \mid \ldots \mid 0)$

and it may be choosen such that all of its entries are positive in Leontief open production super model.

Note this super model yields easy comparison as well as this super model can with different set of price super column vectors and exchange super row matrix find the best solution from the p solutions got from the relation

$(I – A) P = (0 \mid \ldots \mid 0)$
i.e., $((I – A_1) P_1 \mid \ldots \mid (I – A_p) P_p) = (0 \mid \ldots \mid 0)$.

Thus this is also an added advantage of the model. It can study simultaneously p different exchange matrix with p set of price vectors for different industries to study the super interrelations between prices, outputs and demands simultaneously.

Suppose one wants to make a study of interrelation between prices, outputs and demands in an industry with different types of products with different exchange matrix and hence different set of price vectors or of many different industries with same type of products its interrelation between prices, outputs and demands in different locations of the country were the economic status and the education status vary in different locations, how to make a single model to study the situation. In both the cases one can make use of the super input-output model the relation matrix which is a input-output super diagonal mixed square matrix, which will be described presently.

The exchange matrix with p distinct economic models is used to describe the interrelations between prices, outputs and demands. Then the related matrix A will be a super diagonal mixed square matrix

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_p \end{array}\right)$$

$A_1, \ldots, A_p$ are the exchange matrices describing the p-economic models. Now A acts as integrated models in which all the p entities function simultaneously. Now any price vector P will be a super mixed column matrix

$$P = \left(\begin{array}{c} P_1 \\ \hline \vdots \\ \hline P_p \end{array}\right)$$

where each

$$P_t = \left(\begin{array}{c} p_1^t \\ \hline \vdots \\ \hline p_{n_t}^t \end{array}\right);$$

for t = 1, 2, …, p.

Here each A_t is a $n_t \times n_t$ exchange matrix; t = 1, 2, …, p. AP = P is given by

$$A = \left(\begin{array}{c|c|c|c} A_1 & 0 & & 0 \\ \hline 0 & A_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_p \end{array}\right),$$

$$P = \left(\begin{array}{c} P_1 \\ \hline \vdots \\ \hline P_p \end{array}\right)$$

$$AP = \left(\begin{array}{c|c|c|c} A_1P_1 & 0 & & 0 \\ \hline 0 & A_2P_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & A_pP_p \end{array}\right) = \left(\begin{array}{c} P_1 \\ \hline \vdots \\ \hline P_p \end{array}\right)$$

i.e. $A_tP_t = P_t$ for every t = 1, 2, …, p. i.e.

$$\left(\begin{array}{c|c|c|c} (I_1 - A_1)P_1 & 0 & & 0 \\ \hline 0 & (I_2 - A_2)P_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & (I_n - A_n)P_n \end{array}\right)$$

$$= \left(\begin{array}{c|c|c|c} 0 & 0 & & 0 \\ \hline 0 & 0 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & 0 \end{array}\right).$$

Thus AP = P has a nontrivial solution

$$P = \left(\begin{array}{c} P_1 \\ \hline \vdots \\ \hline P_p \end{array}\right)$$

whose entries in each P_t are non negative; $1 \leq t \leq p$.

Let A be the super exchange diagonal mixed square matrix such that for some p-tuple of positive integers $m = (m_1, \ldots, m_p)$, $A_t^{m_t}$ is positive; $1 \leq t \leq p$. Then there is exactly only one linearly independent solution;

$$(I - A)P = \left(\begin{array}{c|c|c|c} 0 & 0 & & 0 \\ \hline 0 & 0 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & 0 \end{array}\right)$$

and it may be choosen such that all of its entries are positive in Leontief open production super model.

Next we proceed on the describe super closed row model (or row closed super model) as the super closed model (or closed super model).

Here we have p sets of K industries which are distributed among themselves i.e. the first set with K industries distributed among themselves, the second set with some K industries distributed among themselves and so on and the p set with some K industries distributed among themselves. It may be that some industries are found in more than one set and some industries in one and only one set and some industries in all the p sets. This open super row model which we choose to call as, when p sets of K industries get distributed among themselves attempts to satisfy an outside demand for outputs.

Portions of these outputs may still be distributed among the industries themselves to keep them operating, but there is to be some excess some net production with which they satisfy the outside demand. In some super closed row models the outputs of the industries in those sets which they belong to were fixed and our objective was to determine sets of prices for these outputs so that the equilibrium condition that expenditure equal income was satisfied for each of the p sets individually.

Thus we will have

x_i^t = monetary value of the total output of the i^{th} industry in the t^{th} set $1 \leq i \leq K$ and $1 \leq t \leq p$.

d_i^t = monetary value of the output of the i^{th} industry of the t^{th} set needed to satisfy the outside demand, $1 \leq t \leq p$, $I = 1, 2, \ldots, K$.

σ_{ij}^t = monetary value of the output of the i^{th} industry needed by the j^{th} industry of the t^{th} set to produce one unit of monetary value of its own output, $1 \leq i \leq K$; $1 \leq t \leq p$.

With these qualities we define the production super column vector

$$X = \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_t \\ \hline \vdots \\ \hline X_p \end{array}\right) = \left(\begin{array}{c} x_1^1 \\ \vdots \\ x_K^1 \\ \hline \vdots \\ \hline x_1^p \\ \vdots \\ x_K^p \end{array}\right).$$

The demand column super vector

$$d = \left(\begin{array}{c} d_1 \\ \hline \vdots \\ \hline d_t \\ \hline \vdots \\ \hline d_p \end{array}\right) = \left(\begin{array}{c} d_1^1 \\ \vdots \\ d_K^1 \\ \hline \vdots \\ \hline d_1^P \\ \vdots \\ d_K^P \end{array}\right)$$

and the consumption super row matrix $C = (C_1 \mid \ldots \mid C_p)$

$$= \left\{ \begin{pmatrix} \sigma_{11}^1 & \sigma_{12}^1 & \ldots & \sigma_{1K}^1 \\ \sigma_{21}^1 & \sigma_{22}^1 & \ldots & \sigma_{2K}^1 \\ \vdots & \vdots & & \vdots \\ \sigma_{K1}^1 & \sigma_{K2}^1 & \ldots & \sigma_{KK}^1 \end{pmatrix} \middle| \ldots \middle| \begin{pmatrix} \sigma_{11}^p & \sigma_{12}^p & \ldots & \sigma_{1K}^p \\ \sigma_{21}^p & \sigma_{22}^p & \ldots & \sigma_{2K}^p \\ \vdots & \vdots & & \vdots \\ \sigma_{K1}^p & \sigma_{K2}^p & \ldots & \sigma_{KK}^p \end{pmatrix} \right\}.$$

By their nature we have

$$X \geq \left(\begin{array}{c} 0 \\ \hline \vdots \\ \hline 0 \end{array}\right); \; d > \left(\begin{array}{c} 0 \\ \hline \vdots \\ \hline 0 \end{array}\right) \text{ and } C > (0 \mid \ldots \mid 0).$$

For the t^{th} set from the definition of σ_{ij}^{t} and x_{j}^{t} it can be seen that the quantity

$$\sigma_{i1}^{t}\,x_{1}^{t} + \sigma_{i2}^{t}\,x_{2}^{t} + \ldots + \sigma_{iK}^{t}\,x_{K}^{t}$$

is the value of the i^{th} industry needed by all the K industries (of the set t) to produce a total output specified by the production vector X_t. Since this quantity is simply the i^{th} entry of the column vector C_tX_t we can further say that the i^{th} entry of the column vector $X_t - X_tC_t$ is the value of the excess output of the i^{th} industry (from the t^{th} set) available to satisfy the outside demand.

The value of the outside demand for the output of the i^{th} industry (from the t^{th} set) is the i^{th} entry of the demand vector d_t; consequently we are lead to the following equation for the t^{th} set $X_t - C_tX_t = d_t$ or $(I - C_t)X_t = d_t$ for the demand to be exactly met without any surpluses or shortages. Thus given C_t and d_t our objective is to find a production vector $X_t \geq 0$ which satisfies the equation

$$(I - C_t)X_t = d_t,$$

so for the all p sets we have the integrated equation to be

$$\begin{gathered} (I - C)X = d \\ \text{i.e., } [(I - C_1)X_1 \mid \ldots \mid (I - C_p)X_p] \\ = (d_1 \mid \ldots \mid d_p)\ . \end{gathered}$$

The consumption super row matrix $C = (C_1 \mid \ldots \mid C_p)$ is said to be super productive if

$$(I - C)^{-1} = [(I - C_1)^{-1} \mid \ldots \mid (I - C_p)^{-1}]$$

exists and

$$(I - C)^{-1} = [(I - C_1)^{-1} \mid \ldots \mid (I - C_p)^{-1}] \geq [0 \mid \ldots \mid 0]\,.$$

A consumption super row matrix is super productive if and only if for some production super vector

$$X = \begin{pmatrix} X_1 \\ \hline \vdots \\ \hline X_n \end{pmatrix} \geq \begin{pmatrix} 0 \\ \hline \vdots \\ \hline 0 \end{pmatrix}$$

such that X > CX i.e. $[X_1 \mid \ldots \mid X_p] > [C_1X_1 \mid \ldots \mid C_pX_p]$.

A consumption super row matrix is productive if each of its row sums is less than one. A consumption super row matrix is productive if each of its column super sums is less than one. The main advantage of this super model is that one can work with p sets of industries simultaneously provided all the p sets have same number of industries (here K). This super row model will help one to monitor and study the performance of an industry which is present in more than one set and see its functioning in each of the sets. Such a thing may not be possible simultaneously in any other model.

Suppose we have p sets of industries and each set has different number of industries say in the first set output of K_1 industries are distributed among themselves. In the second set output of K_2 industries are distributed among themselves so on in the p^{th} set output of K_p-industries are distributed among themselves the super open model is constructed to satisfy an outside demand for the outputs. Here one industry may find its place in one and only one set or group. Some industries may find its place in several groups. Some industries may find its place in every group. To construct a closed super model to analyze the situation.

Portions of these outputs may still be distributed among the industries themselves to keep them operating, but there is to be some excess some net production with which to satisfy the outside demand.

Let

X_i^t	=	monetary value of the total output of the i^{th} industry in the t^{th} set (or group).
d_i^t	=	monetary value of the output of the i^{th} industry of the group t needed to satisfy the outside demand.
σ_{ij}^t	=	monetary value of the output of the i^{th} industry needed by the j^{th} industry to produce one unit monetary value of its own output in the t^{th} set or group, $1 \leq t \leq p$.

With these qualities we define the production super mixed column vector

$$X = \begin{pmatrix} X_1 \\ \hline \vdots \\ \hline X_t \\ \hline \vdots \\ \hline X_p \end{pmatrix} = \begin{pmatrix} x_1^1 \\ \vdots \\ x_{K_1}^1 \\ \hline \vdots \\ \hline x_1^p \\ \vdots \\ x_{K_p}^p \end{pmatrix}$$

and the demand super mixed column vector

$$d = \begin{pmatrix} d_1 \\ \hline \vdots \\ \hline d_p \end{pmatrix} = \begin{pmatrix} d_1^1 \\ \vdots \\ d_{K_1}^1 \\ \hline \vdots \\ \hline d_1^p \\ \vdots \\ d_{K_p}^p \end{pmatrix}$$

and the consumption super diagonal mixed square matrix

$$C = \left(\begin{array}{c|c|c|c} C_1 & 0 & & 0 \\ \hline 0 & C_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & C_p \end{array}\right)$$

where

$$C_t = \begin{pmatrix} \sigma^t_{11} & \sigma^t_{12} & \cdots & \sigma^t_{1K_t} \\ \sigma^t_{21} & \sigma^t_{22} & \cdots & \sigma^t_{2K_t} \\ \vdots & \vdots & & \vdots \\ \sigma^t_{K_t1} & \sigma^t_{K_t2} & \cdots & \sigma^1_{K_tK_t} \end{pmatrix};$$

true for t = 1, 2, …, p.

By the nature of the closed model we have

$$X = \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_p \end{array}\right) = \left(\begin{array}{c} 0 \\ \hline \vdots \\ \hline 0 \end{array}\right),\ d = \left(\begin{array}{c} d_1 \\ \hline \vdots \\ \hline d_p \end{array}\right) = \left(\begin{array}{c} 0 \\ \hline \vdots \\ \hline 0 \end{array}\right)$$

and

$$C = \left(\begin{array}{c|c|c|c} C_1 & 0 & & 0 \\ \hline 0 & C_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & C_p \end{array}\right) = \left(\begin{array}{c|c|c|c} 0 & 0 & & 0 \\ \hline 0 & 0 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & 0 \end{array}\right).$$

From the definition of σ^t_{ij} and x^t_j for every group (set t) it can be seen the quantity $\sigma^t_{ij} X^t_1 + \ldots \sigma_{i_{Kt}} X^t_{K_t}$ is the value of the output of the ith industry needed by all K_t industries (in the tth group) to produce a total output specified by the production vector X_t ($1 \leq t \leq p$). Since this quantity is simply the i^{th} entry of the super column vector in

$$CX = \left(\begin{array}{c|c|c|c} C_1 & 0 & & 0 \\ \hline 0 & C_2 & & 0 \\ \hline & & & \\ \hline 0 & 0 & & C_p \end{array}\right)_{p\times p} \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_p \end{array}\right)_{p\times 1}$$

$$= [C_1X_1 \mid \ldots \mid C_pX_p]^t$$

we can further say that the i^{th} entry of the super column vector $X_t - CX_t$ in

$$X - CX = \left(\begin{array}{cc} X_1 - & C_1X_p \\ \hline \vdots & \vdots \\ \hline X_p - & C_pX_p \end{array}\right)$$

is the value of the excess output of the i^{th} industry available to satisfy the output demand.

The value of the outside demand for the output of the i^{th} industry (in t^{th} set / group) is the i^{th} entry of the demand vector d_t; consequently we are led to the following equation

$$X_t - C_tX_t = d_t \text{ or } (I_t - C_t)\, X_t = d_t\,,\ (1 \leq t \leq p),$$

for the demand to be exactly met without any surpluses or shortages.

Thus given C_t and d_t our objective is to find a production vector $X_t \geq 0$ which satisfy the equation $(I_t - C_t)X_t = d$.

The integrated super model for all the p-sets (or groups) is given by $X - CX = d$ i.e.,

$$\left(\begin{array}{c} X_1 - C_1X_1 \\ \hline X_2 - C_2X_2 \\ \hline \vdots \\ \hline X_p - C_pX_p \end{array}\right) = \left(\begin{array}{c} d_1 \\ \hline d_2 \\ \hline \vdots \\ \hline d_p \end{array}\right)$$

or

$$\left(\begin{array}{c|c|c|c} (I_1 - C_1) & 0 & & 0 \\ \hline 0 & I_2 - C_2 & & \\ \hline & & & \\ \hline 0 & 0 & & I_p - C_p \end{array}\right) \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_p \end{array}\right) = \left(\begin{array}{c} d_1 \\ \hline \vdots \\ \hline d_p \end{array}\right)$$

i.e.,

$$\left(\begin{array}{c} (I_1 - C_1)X_1 \\ \hline \vdots \\ \hline (I_p - C_p)X_p \end{array}\right) = \left(\begin{array}{c} d_1 \\ \hline \vdots \\ \hline d_p \end{array}\right)$$

where I is a $K_t \times K_t$ square identity matrix t = 1, 2, …, p.

Thus given C and d our objective is to find a production super column mixed vector

$$X = \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_p \end{array}\right) \geq \left(\begin{array}{c} 0 \\ \hline \vdots \\ \hline 0 \end{array}\right)$$

which satisfies equation $(I - C)X = d$

$$\text{i.e.} \quad \left(\begin{array}{c} (I_1 - C_1)X_1 \\ \hline \vdots \\ \hline (I_p - C_p)X_p \end{array}\right) = \left(\begin{array}{c} d_1 \\ \hline \vdots \\ \hline d_p \end{array}\right).$$

A consumption super diagonal matrix C is productive if $(I - C)^{-1}$ exists and i.e.,

$$\left(\begin{array}{c|c|c|c} (I_1 - C_1)^{-1} & 0 & & 0 \\ \hline 0 & (I_2 - C_2)^{-1} & & 0 \\ \hline & & & \\ \hline 0 & & & (I_p - C_p)^{-1} \end{array}\right)$$

exists and

$$\left(\begin{array}{c|c|c|c} (I_1 - C_1)^{-1} & 0 & & 0 \\ \hline 0 & (I_2 - C_2)^{-1} & & 0 \\ \hline & & & \\ \hline 0 & & & (I_p - C_p)^{-1} \end{array}\right) \geq$$

$$\left(\begin{array}{c|c|c|c} 0 & 0 & & 0 \\ \hline 0 & 0 & & 0 \\ \hline & & & \\ \hline 0 & & & 0 \end{array}\right).$$

A consumption super diagonal matrix C is super productive if and only if there is some production super vector

$$X = \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_p \end{array}\right) \geq \left(\begin{array}{c} 0 \\ \hline \vdots \\ \hline 0 \end{array}\right)$$

such that

$$X > CX \text{ i.e. } \left(\begin{array}{c} X_1 \\ \hline \vdots \\ \hline X_p \end{array}\right) > \left(\begin{array}{c} C_1X_1 \\ \hline \vdots \\ \hline C_pX_p \end{array}\right).$$

A consumption super diagonal mixed square matrix is productive if each row sum in each of the component matrices is less than one. A consumption super diagonal mixed square matrix is productive if each of its component matrices column sums is less than one.

The main advantage of this system is this model can study different sets of industries with varying strength simultaneously. Further the performance of any industry which is present in one or more group can be studied and also analysed. Such comprehensive and comparative study can be made using these super models.

Finally we develop the notion of super special codes.

Here we define two new classes of super special row codes using super row matrix and super mixed row matrix.

DEFINITION 4.18: *Suppose we have to transform some n set of $k_1, \ldots, k_n$ message symbols $a_1^1 a_2^1 \ldots a_{k_1}^1$, $a_1^2 a_2^2 \ldots a_{k_2}^2$, …, $a_1^n a_2^n \ldots a_{k_n}^n$, $a_i^t \in F_q$; $1 \le t \le n$ and $1 \le i \le k_i$ (q a power of a prime) as a set of code words simultaneously into n-code words such that each code word is of length n_i, i = 1, 2, …, n and $n_1 - k_1 = n_2 - k_2 = \ldots = n_n - k_n = m$ say i.e., the number check symbols of every code word is the same i.e., the number of message symbols and the length of the code word may not be the same. That is the code word consisting of n code words can be represented as a super row vector;*

$$x_s = \left[x_1^1 x_2^1 \ldots x_{n_1}^1 \;\middle|\; x_1^2 x_2^2 \ldots x_{n_2}^2 \;\middle|\; \ldots \;\middle|\; x_1^n x_2^n \ldots x_{n_n}^n \right]$$

$n_i > k_i$, $1 \le i \le n$. In this super row vector $x_j^i = a_j^i$, $1 \le j \le k_i$; i = 1, 2, …, n and the remaining $n_i - k_i$ elements $x_{k_i+1}^i \, x_{k_i+2}^i \ldots x_{n_i}^i$ are check symbols or control symbols; i = 1, 2, …, n.

These n code words denoted collectively by x_s will be known as the super special row code word.

As in case of usual code, the check symbols can be obtained in such a way that the super special code words x_s satisfy a super system of linear equations; $H_s x_s^T = (0)$ where H_s is a super mixed row matrix given by $H_s = [H_1 | H_2 | \ldots | H_n]$ where each H_i is a $m \times n_i$ matrix with elements from F_q, i = 1, 2, …, n, i.e.,

$$H_s x_s^T = [H_1 | H_2 | \ldots | H_n] \left[x_s^1 \; x_s^2 \; \ldots \; x_s^n\right]^T$$
$$= \left[H_1 \left(x_s^1\right)^T \middle| H_2 \left(x_s^2\right)^T \middle| \ldots \middle| H_2 \left(x_s^n\right)^T\right]$$
$$= [|(0) | (0) | \ldots | (0)]$$

i.e., each H_i is the partity check matrix of the code words x_s^i; i = 1, 2, …, n. $H_s = [H_1 | H_2 | \ldots | H_s]$ will be known as the super special parity check super special matrix of the super special row code C_s. C_s will also be known as the linear $[(n_1\ n_2 \ldots n_n), (k_1\ k_2 \ldots k_n)]$ or $[(n_1, k_1), (n_2, k_2), \ldots, (n_n, k_n)]$ super special row code.

If each of the parity check matrix H_i is of the form $\left(A_i, I_{n_i - k_i}\right)$; i = 1, 2, …, n.

$$H_s = [H_1 | H_2 | \ldots | H_n]$$
$$= \left[\left(A_1, I_{n_1-k_1}\right) \middle| \left(A_2, I_{n_2-k_2}\right) \middle| \ldots \middle| \left(A_n, I_{n_n-k_n}\right)\right] \quad \text{----} \quad \text{(I)}$$

C_s is then also called a systematic linear $((n_1\ n_2 \ldots n_n), (k_1\ k_2 \ldots k_n))$ super special code.

If q = 2 then C_s is a super special binary row code; $(k_1 + \ldots + k_n) \mid (n_1 + n_2 + \ldots + n_n)$ is called the super transmission(or super information) rate.

It is important and interesting to note the set C_s of solutions x_s of $H_s x_s^T = (0)$ i.e., known as the super solution space of the super system of equations. Clearly this will form the super special subspace of the super special vector space over F_q of super special dimension ($k_1\ k_2 \ldots k_n$).

C_s being a super special subspace can be realized to be a group under addition known as the super special group code, where H_s is represented in the form given in equation I will be known as the standard form.

Now we will illustrate this super special row codes by some examples.

Example 4.1: Suppose we have a super special binary row code given by the super special parity check matrix $H_S = [H_1 \mid H_2 \mid H_3]$ where

$$H_1 = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$H_2 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$H_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

i.e., the super row matrix associated with the super special code is given by

$$H_s = \left[\begin{array}{cccccc|ccccccc|cccccccc} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right]$$

$$= [(A_1, I_3) \mid (A_2, I_3) \mid (A_3, I_3)] ;$$

i.e., we have given the super special code which is a binary row code. The super special code words are given by

$$x_s = \left[a_1^1\ a_2^1\ a_3^1\ x_4^1\ x_5^1\ x_6^1 \mid a_1^2\ a_2^2\ a_3^2\ a_4^2\ x_5^2\ x_6^2\ x_7^2 \mid a_1^3\ a_2^3\ a_3^3\ a_4^3\ a_5^3\ x_6^3\ x_7^3\ x_8^3\right] = \left[x_s^1 \mid x_s^2 \mid x_s^3\right].$$

$H_s x_s^T$ = (0) gives 3 sets of super linear equations i.e., $H_s x_s^T$ = (0) is read as

$$[H_1 \mid H_2 \mid H_3] \left[x_s^1 \mid x_s^2 \mid x_s^3\right]^T = \left[H_1\left(x_s^1\right)^T \mid H_2\left(x_s^2\right)^T \mid H_3\left(x_s^3\right)^T\right]$$
$$= [(0) \mid (0) \mid (0)];$$

i.e.,

$$H_1\left(x_s^1\right)^T = (0)$$

results in linear equations solving which we can get the code words.

Thus the number of super special row code words in this example of the super special code C_s is 8 × 16 × 32.

The super transmission rate is 12/21. Thus this code has several advantages which will be enumerated in the last chapter of this book. We give yet another example of super special code in which every super special code word is a super row vector and not a super mixed row vector.

Now we proceed on to define the notion of super special row repetition code.

DEFINITION 4.19: *Let* $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$ *be a super special row code in which each of the* C_s^i *is a repetition code, i*

= 1, 2, ..., n, then we define C_s to be a super special repetition row code. Here if $H_s = [H_1|H_2| \ldots |H_n]$ is the super special parity check matrix of C_s, then each H_i is a $t - 1 \times t$ matrix that is we have

$$H_1 = H_2 = \ldots = H_n = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & 0 & 0 & \ldots & 1 \end{bmatrix}_{t-1 \times t}$$

is the parity check matrix. The super special code words associated with C_s are just super row vectors only and not super mixed row vectors. The number of super special code words in C_s is 2^n.

We illustrate a super special row repetition code by the following example.

Example 4.2: Let $C_s = \left[C_s^1 \mid C_s^2 \mid C_s^3 \mid C_s^4 \right]$ be a super row repetition code with associated super special row matrix H_s = $[H_1 \mid H_2 \mid H_3 \mid H_4]$

$$= \left[\begin{array}{ccccc|ccccc} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right.$$

$$\left.\begin{array}{|ccccc|ccccc} 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right].$$

DEFINITION 4.20: *Let C_s be a super special parity check mixed row code i.e., $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n \right]$ where C_s is obtained* *using the super special mixed row matrix $H_s = [H_1 \mid H_2 \mid \ldots \mid H_n]$ where each H_i is a unit row vector having t_i number of elements i.e.,*

$$H_s = \left[\underbrace{1 \;\; 1 \;\; \cdots \;\; 1}_{t_1 \text{ times}} \;\middle|\; \underbrace{1 \;\; 1 \;\; \cdots \;\; 1}_{t_2 \text{ times}} \;\middle|\; \cdots \;\middle|\; \underbrace{1 \;\; 1 \;\; \cdots \;\; 1}_{t_n \text{ times}}\right]$$

where at least one $t_i \neq t_j$ for $i \neq j$. Any super special code word in C_s would be of the form

$$x_s = \left[x_1^1\, x_2^1 \ldots x_{t_1}^1 \;\middle|\; x_1^2 x_2^2 \ldots x_{t_2}^2 \;\middle|\; \ldots \middle| x_1^n x_2^n \ldots x_{t_n}^n\right] = \left[x_s^1 \middle| x_s^2 \middle| \ldots \middle| x_s^n\right]$$

with $H_s x_s^T = (0)$; i.e., each x_s^i would contain only even number of ones and the rest are zeros.

$C_s = [C_1 \mid C_2 \mid \ldots \mid C_n]$ is defined to be super special parity check row code. C_s is obtained from the parity check row matrix / vector $H_s = [H_1 \mid H_2 \mid \ldots \mid H_n]$ where $H_1 = H_2 = \ldots = H_n$ $= \left[\underbrace{1 \;\; 1 \;\; \cdots \;\; 1}_{m \text{ times}}\right]$. Here a super special codeword in C_s would be a super row vector of the form $\left[x_s^1 \mid x_s^2 \mid \ldots \mid x_s^n\right]$ with each $x_s^i = \left[x_1^i\; x_2^i \;\ldots\; x_m^i\right]$ where only even number of x_j^i are ones and the rest zero, $1 \leq j \leq m$ and $i = 1, 2, \ldots, n$.

Now we will illustrate the two types of super special parity check (mixed) row codes.

Example 4.3: Let $C_s = \left[C_s^1 \middle| C_s^2 \middle| C_s^3\right]$ be a super special parity check code having $H_s = [H_1 \mid H_2 \mid H_3] = [1\ 1\ 1\ 1\ 1 \mid 1\ 1\ 1 \mid 1\ 1\ 1\ 1\ 1]$ to be the super special parity check matrix associated with it.

DEFINITION 4.21: *Let $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$ be a super special row code. Suppose $x_s = \left[x_s^1 \mid x_s^2 \mid \ldots \mid x_s^n\right]$ is a* *transmitted super code word and* $y_s = \left[y_s^1 \mid y_s^2 \mid \ldots \mid y_s^n\right]$ *is the received supercode word then* $e_s = y_s - x_s = \left[y_s^1 - x_s^1 \mid y_s^2 - x_s^2 \mid \ldots \mid y_s^n - x_s^n\right] = \left[e_s^1 \mid e_s^2 \mid \ldots \mid e_s^n\right]$ *is called the super error word or the super error vector.*

We first illustrate how the super error is determined.

Example 4.4: Let $C_s = \left[C_s^1 \mid C_s^2 \mid C_s^3 \mid C_s^4\right]$ be a super special code with associated super parity check row matrix $H_s = [H_1 \mid H_2 \mid H_3 \mid H_4]$

$$= \left[\begin{array}{cccccc|ccccccc|ccccc|cccccccc} 1&0&0&1&0&0&1&0&1&0&1&0&0&1&0&1&0&0&1&0&1&0&0&1&0&0 \\ 1&1&0&0&1&0&0&1&0&1&0&1&0&1&1&0&1&0&1&1&0&1&0&0&1&0 \\ 0&1&0&0&0&1&1&1&0&0&0&0&1&0&1&0&0&1&0&1&1&0&1&0&0&1 \end{array}\right].$$

Clearly $y_s + e_s = x_s$.

DEFINITION 4.22: *The super Hamming distance* $d_s(x_s, y_s)$ *between two super row vectors of the super special vector space* V_s*, where* $x_s = \left[x_s^1 \mid x_s^2 \mid \ldots \mid x_s^n\right]$ *and* $y_s = \left[y_s^1 \mid y_s^2 \mid \ldots \mid y_s^n\right]$ *is the number of coordinates in which* x_s^i *and* y_s^i *differ for i = 1, 2, …, n. The super Hamming weight* $w_s(x_s)$ *of the super vector* $x_s = \left[x_s^1 \mid x_s^2 \mid \ldots \mid x_s^n\right]$ *in* V_s *is the number of non zero coordinates in each* x_s^i*; i = 1, 2, …, n. In short* $w_s(x_s) = d(x_s, (0))$.

As in case of usual linear codes we define super minimum distance d_{min}^s of a super special linear row code C_s as

$$d^{s}_{\min} = \min_{\substack{u_s, \nu_s \in C_s \\ u_s \neq \nu_s}} d_s(u_s, \nu_s),$$

$$d_s(u_s, \nu_s) = d_s(u_s - \nu_s, (0)) = w_s(u_s - \nu_s).$$

Thus the super minimum distance of C_s is equal to the least super weights of all non zero super special code words.

Now the value of

$$d^{s}_{\min} = \min_{\substack{u_s, \nu_s \in C_s \\ u_s \neq \nu_s}} d_s(u_s, \nu_s)$$

$$= \min_{\substack{u_s, \nu_s \in C_s \\ u_s \neq \nu_s}} d_s\left(\left[u^1_s \middle| u^2_s \middle| \ldots \middle| u^n_s\right], \left[v^1_s \middle| v^2_s \middle| \ldots \middle| v^n_s\right]\right)$$

$$= \min \left[d\left(u^1_s,\ \nu^1_s\right) + d\left(u^2_s,\ \nu^2_s\right) + \ldots + d\left(u^n_s,\ \nu^n_s\right)\right].$$

Now $d^{s}_{\min}$ of the super special row code given in example 3.1.2 is 7 verified using the fact in $C_s = \left[C^1_s \mid C^2_s \mid C^3_s\right]$; $d^{s}_{\min} C^1_s$ = 3, $d^{s}_{\min} C^2_s = 2$ and $d^{s}_{\min} C^3_s = 2$. Hence $d^{s}_{\min}\left(C_s\right)$ = 3 + 2 + 2 = 7. So we will denote $d^{s}_{\min} = \min\ d_s\left(u_s, \nu_s\right)$ by $d^{s}_{\min}\left(C_s\right)$, u_s, ν_s $\in C_s$, $u_s \neq \nu_s$.

$$d^{S}_{\min}\left[C^1_s \mid C^2_s \mid \ldots \mid C^n_s\right] = d^{S}_{\min}\left[\left(C^1_s\right) + \left(C^2_s\right) + \ldots + \left(C^n_s\right)\right]$$

$$= \min_{\substack{x^1_s, y^1_s \in C^1_s \\ x^1_s \neq y^1_s}} d\left(x^1_s,\ y^1_s\right) + \min_{\substack{x^2_s, y^2_s \in C^2_s \\ x^2_s \neq y^2_s}} d\left(x^2_s,\ y^2_s\right) + \ldots + \min_{\substack{x^n_s, y^n_s \in C^n_s \\ x^n_s \neq y^n_s}} d\left(x^n_s,\ y^n_s\right).$$

Now we proceed on to define the dual of a super special row code.

DEFINITION 4.23: *Let $C_s = \left[C^1_s \mid C^2_s \mid \ldots \mid C^n_s\right]$ be a super special row $[(n_1, \ldots, n_n), (k_1, \ldots, k_n)]$ binary code. The super special dual row code of C_s denoted by*

$$C^{\perp}_s = \left[\left(C^1_s\right)^{\perp} \mid \left(C^2_s\right)^{\perp} \mid \ldots \mid \left(C^n_s\right)^{\perp}\right]$$

where $\left(C_s^i\right)^{\perp} = \{u_s^i \mid u_s^i \cdot v_s^i = 0 \text{ for all } v_s^i \in C_s^i\}$, *i = 1, 2, …, n. Since in* C_s *we have* $n_1 - k_1 = n_2 - k_2 = \ldots = n_n - k_n$ *i.e., the number of check symbols of each and every code in* C_s^i *is the same for i = 1, 2, …, n. Thus we see* $n = 2k_i$ *alone can give us a dual, in all other cases we will have problem with the compatibility for the simple reason the dual code of* C_s^i *being the orthogonal complement will have* $n_i - k_i$ *to be the dimension, where as* C_s^i *will be of dimension* k_i*, i = 1, 2, …, n. Hence we can say the super special dual code would be defined if and only if* $n_i = 2k_i$ *and such that* $n_1 = n_2 = \ldots = n_n$.

We can define the new notion of super special syndrome to super code words of a super special row code which is analogous to syndrome of the usual codes.

DEFINITION 4.24: *Let* C_s *be a super special row code. Let* H_s *be the associated super special parity check matrix of* C_s *the super special syndrome of any element* $y_s \in V_s$ *where* C_s *is a super special subspace of the super special vector space* V_s *is given by* $S(y_s) = H_s y_s^T$. $S(y_s) = (0)$ *if and only if* $y_s \in C_s$.

Thus this gives us a condition to find out whether the received super code word is a right message or not. Suppose y_s *is the received super special code word, we find* $S(y_s) = H_s y_s^T$ *; if* $S(y_s) = (0)$ *then we accept* y_s *as the correct message if* $S(y_s) = H_s y_s^T \neq (0)$ *then we can declare the received word has error.*

We can find the correct word by the following method. Before we give this method we illustrate how the super special syndrome is calculated.

Example 4.5: Let $C_s = \left[C_s^1 \mid C_s^2 \mid C_s^3 \mid C_s^4\right]$ be a super special row code. Let $H_s = [H_1 \mid H_2 \mid H_3 \mid H_4]$ be the super special parity check matrix of C_s.

Let

$$H_s = \left[\begin{array}{cccccc|cccccc|ccccc|ccccccc} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right].$$

Now we have to shown how to find whether the received super special code word is correct or otherwise. It is important to note that what ever be the super special code word $x_s \in C_s$ (i.e., it may be a super special mixed row vector or not) but the syndrome $S(x_s) = H_s x_s^T$ is always a super special row vector which is not mixed and each row vector is of length equal to the number of rows of H_s.

Now we know that every super special row code C_s is a subgroup of the super special vector space V_s over $Z_2 = \{0, 1\}$. Now we can for any $x_s \in V_s$ define super special cosets as

$$x_s + C_s = \{x_s + c_s \mid c_s \in C_s\}.$$

Thus

$$V_s = \{Z_2 \times Z_2 \times \ldots \times Z_2 \mid Z_2 \times \ldots \times Z_2 \mid \ldots \mid Z_2 \times Z_2 \times \ldots \times Z_2\}$$
$$= C_s \cup [x_s^1 + C_s] \cup \ldots \cup [x_s^t + C_s]$$

where $$C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$$

and

$$x_s = \left[x_1^1 \ldots x_{n_1}^1 \middle| x_1^2 \ldots x_{n_2}^2 \middle| \ldots \middle| x_1^n \ldots x_{n_n}^n\right]$$
$$= \left[x_s^1 \mid x_s^2 \mid \ldots \mid x_s^n\right]$$

and

$$x_s + C_s = \left[x_s^1 + C_s^1 \mid x_s^2 + C_s^2 \mid \ldots \mid x_s^n + C_s^n\right].$$

Now we can find the coset leader of every $x_s^i + C_s^i$ as in case of usual codes described in chapter one of this book. Now if

$$y_s = \left[y_s^1 \mid y_s^2 \mid \ldots \mid y_s^n\right]$$

is the received message and $\left[e_s^1 + (0) \mid e_s^2 + (0) \mid \ldots \mid e_s^n + (0)\right]$ is a special super coset leaders then using the relation $y_s - e_s$ we get $y_s - e_s$ to be super special corrected code word. It is interesting to note that each e_s^i + (0) has a stipulated number of coset leaders depending on n_i, i = 1, 2, …, n.

We will illustrate this by the following example.

Example 4.6: Let $C_s = \left[C_s^1 \mid C_s^2\right]$ be a super special row code. Suppose $H_s = [H_1 \mid H_2]$ be the super special row matrix associated with C_s. Let

$$H_s = [H_1 \mid H_2] = \left[\begin{array}{cccc|ccccc} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \end{array}\right].$$

Now $C_s = \left[C_s^1 \mid C_s^2\right]$ with C_s^1 = {(0 0 0 0), (1 0 1 1), (0 1 0 1), (1 1 1 0)}and C_s^2 = {(0 0 0 0 0), (1 0 0 1 0), (0 1 0 0 1), (0 0 1 1 1), (1 1 0 1 1), (0 1 1 1 0), (1 0 1 0 1), (1 1 1 0 0)}.

C_s = {[0 0 0 0 | 0 0 0 0 0], [1 0 1 1 | 0 0 0 0 0],
[0 1 0 1 | 0 0 0 0 0], [1 1 1 0 | 0 0 0 0 0], [0 0 0 0 | 1 0 0 1 0],
[0 0 0 0 | 0 1 0 0 1], [1 1 1 0 | 0 1 0 0 1], [0 1 0 1 | 0 1 0 0 1],
[1 0 1 1 | 1 0 0 0 1], [0 0 0 0 | 0 0 1 1 1], [1 1 1 0 | 0 0 1 1 1],
[0 1 0 1 | 0 0 1 1 1], [1 0 1 1 | 0 0 1 1 1], [0 0 0 0 | 1 1 0 1 1],
[1 1 1 0 | 1 1 0 1 1], [1 0 1 1 | 1 1 0 1 1], [0 1 0 1 | 1 1 0 1 1],
[0 0 0 0 | 0 1 1 1 0], [1 0 1 1 | 0 1 1 1 0], [0 1 0 1 | 0 1 1 1 0],
[0 0 0 0 | 1 0 1 0 1], [1 0 1 1 | 1 0 1 0 1], [0 1 0 1 | 1 0 1 0 1],
[0 0 0 0 | 1 1 1 0 0], [1 1 1 0 | 1 1 1 0 0], [1 0 1 1 | 1 1 1 0 0],

[0 1 0 1 | 1 1 1 0 0] and so on}.

Clearly $|C_s| = 32$. Now the coset table of C_s^1 is given by

Message	code		words
0 0	1 0	0 1	1 1
0 0 0 0	1 0 1 1	0 1 0 1	1 1 1 0
Other cosets			
1 0 0 0	0 0 1 1	1 1 0 1	0 1 1 0
0 1 0 0	1 1 1 1	0 0 0 1	1 0 1 0 .
0 0 1 0	1 0 0 1	0 1 1 1	1 1 0 0
coset leaders			

Now the coset table of C_s^2 is given by

message	0 0 0	1 0 0
codewords	0 0 0 0 0	1 0 0 1 0
other cosets	1 0 0 0 0	0 0 0 1 0
	0 1 0 0 0	1 1 0 1 0
	0 0 1 0 0	1 0 1 1 0

message	0 1 0	0 0 1
codewords	0 1 0 0 1	0 0 1 1 1
other cosets	1 1 0 0 1	1 0 1 1 1
	1 0 0 0 1	0 1 1 1 1
	0 1 1 0 1	0 0 0 1 1

$$\begin{array}{lccccc|ccccc}
\text{message} & 1 & 1 & 0 & & & 0 & 1 & 1 & & \\
\text{codewords} & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 \\
\text{other cosets} \left\{ \begin{array}{l} \\ \\ \\ \end{array} \right. & \begin{array}{c}0\\1\\1\end{array} & \begin{array}{c}1\\0\\1\end{array} & \begin{array}{c}0\\0\\1\end{array} & \begin{array}{c}1\\1\\1\end{array} & \begin{array}{c}1\\1\\1\end{array} & \begin{array}{c}1\\0\\0\end{array} & \begin{array}{c}1\\0\\1\end{array} & \begin{array}{c}1\\1\\0\end{array} & \begin{array}{c}1\\1\\1\end{array} & \begin{array}{c}0\\0\\0\end{array}
\end{array}$$

$$\begin{array}{lccccc|ccccc}
\text{message} & 1 & 0 & 1 & & & 1 & 1 & 1 & & \\
\text{codewords} & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\
\text{other cosets} \left\{ \begin{array}{l} \\ \\ \\ \end{array} \right. & \begin{array}{c}0\\1\\1\end{array} & \begin{array}{c}0\\1\\0\end{array} & \begin{array}{c}1\\1\\0\end{array} & \begin{array}{c}0\\0\\0\end{array} & \begin{array}{c}1\\1\\1\end{array} & \begin{array}{c}0\\1\\1\end{array} & \begin{array}{c}1\\0\\1\end{array} & \begin{array}{c}1\\1\\0\end{array} & \begin{array}{c}0\\0\\0\end{array} & \begin{array}{c}0\\0\\0\end{array}
\end{array}.$$

Suppose $y_s = [1\ 1\ 1\ 1 \mid 1\ 1\ 1\ 1\ 1]$ is the received word then $S(y_s) = H_s y_s^T \neq [(0) \mid (0)]$. $e_s = [0\ 1\ 0\ 0 \mid 0\ 0\ 1\ 0\ 0]$ is the super set coset leader. Thus $x_s = y_s + e_s = [1\ 0\ 1\ 1 \mid 1\ 1\ 0\ 1\ 1] \in C_s$.

With the advent of computers calculating the super special coset leaders is not a very tedious job. Appropriate programs will yield the result in no time.

Now we proceed on to describe/define the super special row cyclic code.

DEFINITION 4.25: *Let $C_s = [C_1 \mid C_2 \mid \ldots \mid C_n]$ be a super special row code. If every C_s^i is a cyclic code in C_s we call C_s to be a super special cyclic row code. $H_s = [H_1 \mid H_2 \mid \ldots \mid H_n]$ denotes the super special parity check row matrix of the super special cyclic code.*

We illustrate this by the following example.

Example 4.7: Let $C_s = \left[C_s^1 \mid C_s^2 \mid C_s^3\right]$ be a super special cyclic code with an associated super special parity check matrix

$$H_s = [H_1 \mid H_2 \mid H_3]$$

$$= \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

$$\left. \begin{array}{cccccc|cccccc} 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{array} \right].$$

We see each of C_s^1, C_s^2 and C_s^3 are cyclic codes.

Now we see in general for any super special mixed row code with an associated super special parity check matrix H_s which happens to be a super mixed row matrix we cannot define the super special generator row matrix G_s. The simple reason being if the code words in each of the C_s^i in C_s where $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$, i = 1, 2, …, n happens to be of different length then it would be impossible to define a super generator matrix. So we shall first define the notion of super special generator row matrix of a super special row code(mixed row code).

DEFINITION 4.26: *Let $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$ be a super special row code. A super special row matrix which generates C_s exists if and only if in each C_s^i the codes in C_s have the same number of message symbols, that is if C_s has a super special parity check row matrix $H_s = [H_1 \mid H_2 \mid \ldots \mid H_n]$ then we demanded each C_s^i must have the same number of check symbols. Likewise for the super special generator row matrix to exist we must have $G_s = [G_1 \mid G_2 \mid \ldots \mid G_n]$ where C_s^i have the same number of message symbols which forms the number of rows of the super row generator matrix G_s.*

We shall first illustrate this by an example.

Example 4.8: $C_s = \left[C_s^1 \mid C_s^2\right]$ be a super special row code.

Let $$G_s = \left[G_s^1 \mid G_s^2\right]$$

$$= \left[\begin{array}{cccccc|cccccc} 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

One can easily find the super code words using G_S [50].

DEFINITION 4.27: *Let $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$ be a super special mixed row code. If each of the codes C_s^i have the same number of message symbols then we have the super special generator mixed row matrix $G_s = \left[G_s^1 \mid G_s^2 \mid \ldots \mid G_s^n\right]$ associated with C_s. Number of message symbols in each of the G_s^i are equal and is the super special mixed row matrix G_s; $1 \leq i \leq n$.*

We illustrate this by the following example.

Example 4.9: Let $C_s = \left[C_s^1 \mid C_s^2 \mid C_s^3 \mid C_s^4\right]$ be a super special mixed row code. Let $G_s = \left[G_s^1 \mid G_s^2 \mid G_s^3\right]$ be the associated super special mixed row generator matrix given by

$$G_s = \left[G_s^1 \mid G_s^2 \mid G_s^3\right]$$

$$= \left[\begin{array}{cccccc} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{array}\right.$$

$$\left|\begin{array}{ccccccc|ccccccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{array}\right].$$

THEOREM 4.4: *Let $C_s = \left[C_s^1 \mid C_s^2 \mid \ldots \mid C_s^n\right]$ be a super special row code with $H_s = [H_1 \mid H_2 \mid \ldots \mid H_n]$, the super special parity check matrix. If each $H_i = (A_i, I_{n-k})$, $i = 1, 2, \ldots, n$ then $G_s = [G_1 \mid G_2 \mid \ldots \mid G_n]$ with $G_i = (I_k, -A^T)$; $1 \leq i \leq n$ if and only if the length of each code word in C_s^i is the same for $i = 1, 2, \ldots, n$.*

For proof refer [50].

For any super special row code we can H_S the parity check matrix and the super special generator matrix G_S such that $G_S H_S^T$ = (zero super vector).

Now having defined the new class of super special row (mixed row) codes we will now define new classes of mixed super classes of mixed super special row codes C_s i.e., we may have the super special row code to contain classical subcodes as Hamming code or cyclic code or code and its orthogonal complement and so on [50].

DEFINITION 4.28: *Let $C_s = [C_1 \mid C_2 \mid \ldots \mid C_n]$ be a super special row code. If some of the C_i's are Hamming codes, some C_j's are cyclic codes $i \neq j$, some C_k's are repetition codes and some C_t's are codes and C_p's are dual codes of C_t's ; $1 \leq j, k, t, i, p < n$ then we call C_s to be a mixed super special row code.*

It is important to mention here that even if two types of classical codes are present still we call C_s as a mixed super special row code.

We will illustrate them by the following examples.

Example 4.10: Let $C_s = [C_1 \mid C_2 \mid C_3 \mid C_4]$ be a mixed super special row code. Here C_1 is a Hamming code, C_2 the repetition code, C_3 a code of no specific type and C_4 a cyclic code.

Let the mixed super special parity check matrix H_s associated with C_s be given by

$$H_s = [H_1 \mid H_2 \mid H_3 \mid H_4]$$

$$= \left[\begin{array}{ccccccc|cccc|cccccc|ccccccc} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{array}\right].$$

Example 4.11: Let $C_s = [C_1 \mid C_2 \mid C_3]$ be a mixed super special row code. Let $H_s = [H_1 \mid H_2 \mid H_3]$ be the associated super special parity check mixed row matrix. C_1 is the Hamming code, C_2 any code and C_3 a repetition code.

$$H_s = \left[\begin{array}{cccccccc|ccccccc|ccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

is the mixed super special parity check mixed row matrix for which G_s does not exist.

We define the new notion of super special Hamming row code.

DEFINITION 4.29: *Let* $C_S = \left[C_S^1 \mid C_S^2 \mid \ldots \mid C_S^n\right]$ *where each* C_s^i *is a* $(2^m - 1, 2^m - 1 - m)$ *Hamming code for* $i = 1, 2, \ldots, n$. *Then we call* C_s *to be a super special Hamming row code. If* $H_s = [H_1 \mid H_2 \mid \ldots \mid H_n]$ *be the super special parity check matrix associated with* C_s *we see* H_s *is a super special row matrix having m rows and each parity check matrix* H_i *has m rows and* $2^m - 1$ *columns,* $i = 1, 2, \ldots, n$.

Further the transmission rate can never be equal to ½. If m > 2 then will the transmission rate be always greater than ½ ?

For more examples refer [50].

We define using super column matrices the notion of super column codes which is an innovative means of using super column matrices.

DEFINITION 4.30: *Suppose we have to describe n codes each of same length say m but with varying sets of check symbols by a single matrix. Then we define it using super column matrix as the super code parity check matrix. Let*

$$C_s = \begin{bmatrix} C_1 \\ \hline C_2 \\ \hline \vdots \\ \hline C_m \end{bmatrix}$$

be a set of m codes, $C_1, C_2, \ldots, C_m$ *where all of them have the same length n but have* $n - k_1, n - k_2, \ldots, n - k_m$ *to be the number of check symbols and* $k_1, k_2, \ldots, k_m$ *are the number of message symbols associated with each of the codes* $C_1, C_2, \ldots, C_m$ *respectively.*

Let us consider

$$H^s = \begin{bmatrix} H^1 \\ \hline H^2 \\ \hline \vdots \\ \hline H^m \end{bmatrix}$$

where each H^i is the $n - k_i \times n$ parity check matrix of the code C_i; i = 1, 2, ..., m. We call H^s to be the super special parity check mixed column matrix of C_s and C_s is defined as the super special mixed column code.

The main difference between the super special row code and the super special column code is that in super special row codes always the number of check symbols in every code in C_s is the same as the number of message symbols in C_i and the length of the code C_i can vary where as in the super special column code, we will always have the same length for every code C_i in C_s but the number of message symbols and the number check symbols for each and every code C_i in C_s need not be the same. In case if the number of check symbols in each and every code C_i is the same. Then we call C_s to be a super special column code.

In case when we have varying number of check symbols then we call the code C_s to be a super special mixed column code.

In the case of super special column code $C_s = [C_1 \mid C_2 \mid \ldots \mid C_m]^t$ we see every code C_i in C_s have the same number of message symbols. Thus every code is a (n, k) code. It may so happen that some of the C_i and C_j are identical codes.

For examples refer [50].

DEFINITION 4.31: *Let $C_s = [C_1| \ C_2 \mid \ldots \ |C_n]^t$ where C_i's are codes of same length m. Suppose each C_i is generated by a matrix G_i, i = 1, 2, ..., n, then*

$$G_s = \left[\begin{array}{c} G_1 \\ \hline \vdots \\ \hline G_n \end{array}\right]$$

generates the super special column code C_s. We call G_s the super special generator column matrix which generates C_s. If in each of the codes C_i in C_s, we have same number of message symbols then we call G_s to be a super special generator column matrix; $i = 1, 2, \ldots, n$. If each of the codes C_i's in C_s have different number of message symbols then we call G_s to be a super special generator mixed column matrix.

We say G_s is in the standard form only if each G_i is in the standard form. Further only when G_s is in the standard form and G_s is a super special column matrix which is not a mixed matrix we have H_s the super special parity check column matrix of the same C_s with

$$G_s H_s^T = \left[\begin{array}{c} 0 \\ \hline 0 \\ \hline \vdots \\ \hline 0 \end{array}\right].$$

For examples refer [50].

We proceed onto define classical super special column codes.

DEFINITION 4.32: *Let $C_s = [C_1 \mid C_2 \mid \ldots \mid C_n]^t$ be a super special column code if each of the code C_i is a repetition code of length n then C_s is a super special repetition column code with $C_1 = C_2 = \ldots = C_n$.*

The super special column parity check matrix

$$H_s = \left[\begin{array}{c} H \\ \hline \vdots \\ \hline H \end{array}\right]$$

$$= \left[\begin{array}{cccccc} 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & 0 & 0 & 0 & \ldots & 1 \\ \hline & & & \vdots & & \\ \hline 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & 0 & 0 & 0 & \ldots & 1 \end{array}\right]$$

where

$$\mathrm{H} = \begin{bmatrix} 1 & 1 & 0 & 0 & \ldots & 0 \\ 1 & 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & 0 & 0 & 0 & \ldots & 1 \end{bmatrix}.$$

It is important to note that unlike the super special repetition row code which can have different lengths the super special repetition column code can have only a fixed length and any super special code word

$$x_s = \left[x_s^1 \mid x_s^2 \mid \cdots \mid x_s^n \right]$$

where x_s^j *= (1 1 ... 1), n-times or (0 0 ... 0) n-times only;* $1 \leq j \leq n$.

DEFINITION 4.33: *Let $C_s = [C_1 \mid C_2 \mid ... \mid C_n]^t$ be a super special parity check column code . Let the super special parity check column matrix associated with C_s be given by*

$$H_s = \left[\begin{array}{c} H_1 \\ \hline H_2 \\ \hline \vdots \\ \hline H_n \end{array}\right]$$

Where

$$H_1 = H_2 = ... = H_n = \underbrace{(1\ 1\ ...\ 1)}_{m-times}.$$

Thus we see we cannot get different lengths of parity check codes using the super special column code. However using super special row code we can get super special parity check codes of different lengths.

DEFINITION 4.34: *Let $C_s = [C_1 \mid C_2 \mid ... \mid C_n]^t$ be a super special column code if each of the codes C_i in C_s is a $(2^m - 1, 2^m - 1 - m)$ Hamming code for $i = 1, 2, ... , n$ then we call C_s to be a super special column Hamming code . It is pertinent to mention that each code C_i is a Hamming code of same length; $i = 1, 2, ... , n$.*

DEFINITION 4.35: *$C_s = [C_1 \mid C_2 \mid ... \mid C_n]^t$ is a mixed super special column code if some C_i's are repetition codes of length n some C_j's are Hamming codes of length n, some C_k's parity check codes of length n and others are arbitrary codes, $1 \leq i, j, k \leq n$.*

We illustrate this by the following example.

DEFINITION 4.36: *Let $C_s = [C_1 \mid C_2 \mid ... \mid C_m]^t$ be a super special column code if each of the codes C_i is a cyclic code then we call C_s to be a super special cyclic column code. However* *the length of each code C_i; i = 1, 2, …, n will only be a cyclic code of length n, but the number of message symbols and check symbols can be anything.*

Now we illustrate this by an example.

DEFINITION 4.37: *Let*

$$C(S) = \left[\begin{array}{c|c|c|c} C_1^1 & C_2^1 & \cdots & C_n^1 \\ \hline C_1^2 & C_2^2 & \cdots & C_n^2 \\ \hline \vdots & \vdots & \cdots & \vdots \\ \hline C_1^m & C_2^m & \cdots & C_n^m \end{array}\right]$$

where C_j^i are codes $1 \leq i \leq m$ and $1 \leq j \leq n$. Further all codes $C_1^1, C_1^2, \ldots, C_1^m$ are of same length $C_2^1, C_2^2, \ldots, C_2^m$ are of same length and $C_n^1, C_n^2, \ldots, C_2^m$ are of same length. $C_1^1, C_2^1, \ldots, C_n^1$ have same number of check symbols, $C_1^2, C_2^2, \ldots, C_n^2$ have same number of check symbols and $C_1^m, C_2^m, \ldots, C_n^m$ have same number of check symbols.

We call C(S) to be a super special code. We can have the super parity check matrix

$$H(S) = \left[\begin{array}{c|c|c|c} H_1^1 & H_2^1 & \cdots & H_n^1 \\ \hline H_1^2 & H_2^2 & \cdots & H_n^2 \\ \hline \vdots & \vdots & \cdots & \vdots \\ \hline H_1^m & H_2^m & \cdots & H_n^m \end{array}\right]$$

where H_j^i's are parity check matrices $1 \leq i \leq m$ and $1 \leq j \leq n$. Further $H_1^1, H_2^1, \ldots, H_n^1$ have the same number of rows, $H_1^2, H_2^2, \ldots, H_n^2$ have same number of rows and so on. $H_1^m, H_2^m, \ldots, H_n^m$ have the same number of rows. Likewise $H_1^1, H_1^2, \ldots, H_1^m$ have the same number of columns,

$H_2^1, H_2^2, \ldots, H_2^m$ *have the same number of columns and so on.* $H_n^1, H_n^2, \ldots, H_n^m$ *have same number of columns.*

The notion of super matrices are ingeneously applied to super fuzzy models, super linear algebra and super special codes. Such type of applications is not only innovative but can lead to easy comparison and save the working time.

Chapter Five

INTERVAL MATRICES AND NATURAL CLASS OF INTERVALS

In this chapter first we recall properties of intervals and then define the notion of matrix interval using matrices with entries from R. Also we define the matrix of natural class of intervals and illustrate them with examples we however get the relation between the interval matrix and matrix interval and the relation between the natural class of matrix intervals and the matrix of natural class of intervals. We will call [A, B] an interval where A > B or A < B or A = B or A and B are not even comparable but are bound by some common features. When A and B are integers or modulo integers we get intervals which we call them as natural class of intervals. For if [15, 20] where 15 and 20 $\in$ Z_{25} is the situation when we cannot in any way compare them. So if we replace A and B by matrices of same order we call that interval as matrix interval.

With this rough idea now we proceed onto define the row matrix interval.

By an interval [a, b] we mean all elements x such that a ≤ x ≤ b are in [a b]. [a a] is nothing but a. So when in an interval a = b it collapses to a single point.

Let Q_I = {[a, b] | a, b ∈ Q}.

Let Q_I^+ = {[a, b] | a, b ∈ $Q^+ \cup \{0\}$, Q^+ denotes the set of positive rationals}.

Z_I = {[a, b] | a, b ∈ Z}; denotes the set of all intervals on Z, i.e., the set of positive and negative integers.

Z_I^+ = {[a, b] | a, b ∈ $Z^+ \cup \{0\}$; Z^+ the set of positive integers}.

R_I = {[a, b] | a, b ∈ R} i.e., the collection of all intervals in the set of reals.

R_I^+ = {[a, b] | a, b ∈ $R^+ \cup \{0\}$, R^+ the set of positive reals}.

Z_n^I = {[a, b] | a, b ∈ Z_n}.

Thus we have defined the collection of all intervals of different types. We first make the following observation which is very essential.

We see $Q \subseteq Q_I$ (a = [a, b] ∈ Q_I).

Likewise $Z \subseteq Z_I$, $Q^+ \subseteq Q_I^+$, $Z_n \subseteq Z_n^I$, $R_I \supseteq R$, $R^+ \subseteq R_I^+$.

Thus these which will be known as interval sets are generalized or contain the related sets as subsets. We shall call Q_I the rational intervals, Z_I the integer intervals, R_I the real intervals and Z_n^I the modulo integer intervals.

Further $Z \subseteq Z_I \subseteq Q_I \subseteq R_I$

$Z \subseteq Q \subseteq Q_I \subseteq R_I$, $Z \subseteq Q \subseteq R \subseteq R_I$.

[–7, 3] is an integer interval in Z_I.

[3, 5] is a positive integer interval in Z_I^+.

We see $Z^+ \subseteq Z_I^+ \subseteq Z_I$.

[–5/2, 3/7] is a rational interval in Q_I.

[9/7,5/2] is a positive rational interval in Q_I^+; $Q^+ \subseteq Q_I^+ \subseteq Q_I$.

$[-\sqrt{2}, \sqrt{5}]$ is a real interval in R_I but $[\sqrt{3}, \sqrt{7}]$ is a positive real interval in R_I^+. Thus $R^+ \subseteq R_I^+ \subseteq R_I$. We see [0, 1] [0, 4] and modulo integer intervals of Z_5^I.

It is important and interesting to note Z_n^I has only finite number elements in it.

Infact Z_2^I has only four elements viz. {[0, 0], [1, 1] [0, 1]}.

Z_3^I has only 6 elements viz {[0, 0] [1, 1], [2, 2], [0, 1], [0, 2], [1, 2]}.

Z_4^I has only 10 intervals viz. {[0, 0], [1, 1], [2, 2], [3,3]. [0, 1], [0, 2], [0, 3], [1, 2], [1, 3], [2, 3]}.

Z_5^I has only 15 modulo intervals. {[0, 0], [1, 1], [2, 2], [3, 3], [4, 4], [0, 1], [0, 2], [0, 3], [0, 4], [1, 2], [1, 3], [1, 4], [2, 3], [2, 4], [3, 4]}.

Z_7^I has only 28 modulo intervals.

Thus Z_n^I has only n+ (n–1) + … + 2+1 = $\frac{n(n+1)}{2}$ modulo intervals.

Z_{15} has only {[0, 0], [1, 1], [2, 2], …, [14, 14], [0, 1], [0, 2], …, [0, 14], [1, 2], …, [1, 14], [2, 3], …, [2, 14], …, [13, 14]} = Z_{15}^I .

Number of intervals in Z_{15}^I is 120.

Now we recall the operations on intervals given by |26]

DEFINITIONS [26]

[a, b] = [c, d] if and only if a = c and b = d
[a, b] $\subseteq$ [c, d] if only if c ≤ a ≤ b ≤ d.

[a, b] < [c, d] if and only if b < c. The width of an interval [a, b] is denoted by w([a, b]) = b – a and the magnitute of the interval |[a, b]| = max (|a|, |b|).

Degenerate intervals are those intervals which has zero width, can be identified with real numbers. Thus a = [a, a].

We now proceed onto give the arithmetic operations as given by Kuperman.

If * denotes any one of the symbols +, –, ., | (. Product, / division).

Then the interval [a, b] * [c, d] = $\left\{ x * y \,\middle|\, \begin{matrix} a \le x \le b \\ c \le y \le d \end{matrix} \right\}$ except we do not define [a,b] | [c, d] if o ∈ [c, d]. (This excludes the definition by zero possibility). Alternatively we can define the arithmetic operations on intervals by giving the endpoints of the intervals resulting from the sum, difference, product or quotient of two intervals.

Equivalent to the definition in I we thus we

$[a, b] + [c, d] = [a+c, b+d]$
$[a, b] – [c, d] = [a–d, b–c]$
$[a, b] . [c, d] = [\min (ac, ad, bc, bd), \max (ac, ad, bc, bd)]$
$[a, b] / [c, d] = [a, b] / [1/d, 1/c]$, provided $0 \notin [c, d]$.

For the denominator cannot be zero. Thus in the definition of division c and d are either both positive or both negative. By way of numerical example we thus have

(i) [–2, 3] + [–9, –3] = [–11, 0]
[8, 11] + [5, 7] = [13, 18].
[–3, 8] + [3, 4] = [0, 12]

(ii) [–3, 8] – [–7, –2] = [–3 – (–2), 8+2]
= [–1, 10]

[8, 17] – [–2, 6] = [8 – 6, 17 – (–2)]
= [2, 19]

[2, 7] – [–1, 3] = [2–3, 7–(–1)]
= [–1, 8]

(iii) [3, 5] . [7, 9] = [min {21, 27, 35, 45},
max {21, 27, 35, 45}]
= [21, 45]

[–2, 1] . [2, 5] = [min {–4, 2, 5, –10},
max {–4, 2, 5, –10}]
= [–10, 5]

[–5, 2] . [3, 7] = [min {–15, 6, 14, –35},
max {–15, 6, 14, –35}]
= [–35, 14]

[–2, 7] / [3, 8] = [min {–2/3, –2/8, 7/3, 7/8},
max {–2/3, –2/8, 7/8, 7/3}]
= [–2/3, 7/3]

[–5, 4] / [–2,6] = [min {5/2, –5/6, –4/2, 4/6},
max {5/2, –2, 4/6, –5/6}]
= [–2, 5/2]}

[–8, 9] / [3, 7] = [min {–8/3, 8/7, 3, 9/7},
max {–8/3, 8/7, 3, 9/7}
= [–8/3, 3]

Now we recall the properties of interval arithmetic given by Kooperman.

For more about these notions please refer [26]

We may note that interval arithmetic is a generalization or extension of real arithmetic since [a, a] is a real number and the definitions hold for intervals of this form as well.

Thus laws that hold for interval arithmetic must automatically hold for real arithmetic. But laws that hold for real arithmetic may not always hold for interval arithmetic.

For example, interval arithmetic is associative and commutative with respect to addition and multiplication. Thus, given interval numbers I, J, K we have

(I, J) + K = I + (J + K)
I + J = J + I

But distributive laws does not always hold for interval arithmetic.

[3, 5] ([1, 2] + [–7, 2])
= [3, 5] [1, 2] + [3, 5] [–7, 2]
= [min {3, 5, 6, 10}, max {3, 5, 6, 10}] +
[min {–21, 6, –35, 10}, max {–21, 6, –35, 10}]
= [3, 10] + [–35, 10]
= [–32, 20] = [3, 5] [1, 2] + [3, 5] [–7, 2]

Consider [3, 5] ([1, 2] + [–7, 2])
= [3, 5] [–6, 4] = [min {–18, –30, 12, 20},
max {–18, –30, 12, 20}]
= [–30, 20]

Thus [3, 5] ([1, 2] + [–7, 2]) ≠ [3, 5] [1, 2] + [3, 5] [–7, 2].

It is pertinent to mention here that the distributive laws hold good when the interval is a real number say $\overline{k}$; i.e., the degenerate interval [k, k] of zero with, then it is easy to verify

[k, k] ([a, b] + [c, d]) = [k, k] [a, b] + [k, k] [c, d]
= k([a, b] + [c, d]) = k [a, b] + k [c, d]
= k [a+c, b+d] = [ka, kb] + [kc, kd]
= [k(a+c), k(b+c)] = [ka + kc, kb + kd]
= [ka + kc, kb + kd]

Thus we have
b [a – δa, a + δa] = b ([a, a] + [–δa, δa])
= b[a, a] + b[–δa, δa]
[ba, ba] + [–|b| δ a, |b| δa]
= [ba – |b|δa, ba + |b| δa]
b[a – δa, a + δa] = [ba – |b| δa, ba + |b|δa]

(We may note that δa is non negative by implication of its appearance in [a – δa, a + δa] for in any interval [c, d], c ≤ d).

We now proceed onto recall the concept of rounded interval arithmetic. An important use of interval arithmetic is to determine upper bounds for the errors due to rounding.

In rounded interval arithmetic working in either single or double precision, we regard each real number as an interval. An in the first place we round the lower endpoint down and the upper end point up.

In this way, the maximum possible effect of rounding is taken into account, so that the interval for the final result contains the true value.

In using rounded interval arithmetic it should be clear that the final result can be obtained as sharply as desired, i.e., the width of the final interval can be as small as desired, by using a sufficiently long wordlength. By way of a numerical example. Suppose we multiply two number. a = 0.2310 | 581 and b = 0.8351 | 621 is a decimal coded machine using a precision of wordlength of four decimal digits.

Then, in the first place we write down the intervals containing a and b.

$a \in [0.2310, 0.2311]$ and $b \in [0.8351, 0.8352]$

Then carrying out the multiplication we have
$ab \in [0.2310, 0.2311]\ [0.8351, 0.8352]$
$= [0.1929 \mid 18\ , \ldots,\ 0.1929 \mid 92 \ldots]$
$= [0.1929, 0.1930]$.

Thus, rounded interval arithmetic leads to the following interval which contains ab : [0.1929, 0.1930], the width of this interval being 0.1930 – 0.1929 = 0.0001.

To obtain the product ab more sharply we clearly have to use a longer wordlength for the computation.

The actual rounding down and up can most easily be achieved in machine coding. The lower endpoint should be rounded down so that the rounded value is less than or equal to the value being rounded. Thus if the value being rounded down is positive we need merely truncate while if the value is negative we truncate, examine the truncated portion, and, if this is not zero, we subtract one unit in the least significant position. And the upper end point must be rounded up if a computation is being carried out in rounded interval arithmetic, an interval being said to be properly rounded if the endpoints are rounded as described above.

We may mention that the rounding up and down can be coded in Fortan necessary although not as efficiently.

Thus the final interval obtained by rounded interval arithmetic takes account of the errors due to rounding and contains the true value.

Interval vectors and matrices and vectors and matrices whose elements are interval numbers, the superscript I being used to indicate such a vector or matrix.

Thus given matrices $B = (b_{ij})$ and $C = (c_{ij})$ of order n such that

$b_{ij} \leq c_{ij}$, i, j = 1, 2, …, n, then the interval matrix $A^I = [B, C]$ is defined by

$A^I = [B, C] = \{A = (a_{ij}) \mid b_{ij} \leq a_{ij} \leq c_{ij};\ i, j = 1, 2, \ldots, n\}$. Kuperman [26] calls B the lower endpoint matrix of the interval matrix $A^I = [B, C]$, C the upper endpoint matrix and M = ½ (B+C) the midpoint matrix.

DEFINITION 5.1: *Let $A = (a_1, \ldots, a_n)$ and $B = (b_1, \ldots, b_n)$ $a_i, b_i \in R$ and Z_n or Q or Z or C where A and B are of some order, viz; $1 \times n$. We define the row matrix of natural class of interval as*

$$[A, B] = [(a_1, \ldots, a_n), (b_1, \ldots, b_n)].$$

If each $a_i > b_i$, $1 \leq i \leq n$ we say [A, B] such that A > B if each $a_i < b_i$, $1 \leq i \leq n$ we say the row matrix natural class of interval [A, B] with A < B. For more about properties and working with natural class of intervals refer [57-58].

If some $a_i > b_i$ and some $a_j < b_j$ and some $a_k = b_k$ we cannot say A > B or A < B. We will illustrate the row matrix of natural class of interval or in short row matrix interval by some examples.

Example 5.1: Let X = [(5, 3, –1, 0, 2, 4), (2, 4, 5, 7, –3, 1)] = [A, B] is the row matrix interval clearly A is not comparable with B.

Example 5.2: Let Y = [A, B] = [(5, 2, 0, 7, 9, 6, 10), (7, 3, 4, 5, 9, 10, 8, 12)] be a row matrix interval where A < B.

Example 5.3: Let M = [(8, 3, 4, 9, –2, 7, 18, –9), (2, 1, 3, 5, –10, 2, 9, –14) = [A, B] be the row matrix interval where A > B.

Example 5.4: Let S = [(3, 4, 2, 1, 5), (3, 4, 2, 1, 5)] = [A, B] be the row matrix interval, here A = B. So S = (3, 4, 2, 1, 5). We call this row matrix interval as degenerate interval.

We will now proceed onto define the notion of addition, multiplication, subtraction and division of these row interval matrices.

First we define the notion of order in a row interval matrix.

Let X = [A, B] be a row interval matrix where order of A and B is 1 × n. We define the order of X, the row interval matrix to be 1 × n. Thus X is a 1 × n row interval matrix.

We can add two row interval matrices if and only if they are of same order otherwise addition is not defined; for we cannot add a 1 × 5 row matrix with a 1 × 9 row matrix.

DEFINITION 5.2: *Let* $X = [A, B] = [(a_1, a_2, \ldots, a_n), (b_1, b_2, \ldots, b_n)]$ *and* $Y = [C, D] = [(c_1, c_2, \ldots, c_n), (d_1, d_2, \ldots, d_n)]$ *be any two* $1 \times n$ *row matrix interval. We define* $X + Y = [A, B] + [C, D]$

$$= [A + C, B + D]$$
$$= [(a_1, \ldots, a_n) + (c_1, \ldots, c_n), (b_1, \ldots, b_n) + (d_1, \ldots, d_n)]$$
$$= [(a_1 + c_1, \ldots, a_n + c_n), (b_1 + d_1, \ldots, b_n + d_n)]$$

$X + Y$ *is again a* $1 \times n$ *row matrix interval.*

We will first illustrate this by some examples.

Example 5.5: Let X = [A, B] = [(3, 1, 5, 0, –3, 2), (7, 8, 9, 11, 0, 1)] and Y = [C, D] = [(–5, 2, 3, –7, 8, 10), (–2, 11, 10, –4, 5, –3)] be any two 1 × 6 row matrix intervals.

We see X + Y = [A, B] + [C, D]

= [(3, 1, 5, 0, –3, 2), (7, 8, 9, 11, 0, 1)] +
[(–5, 2, 3, –7, 8, 10), (–2, 11, 10, –4, 5, –3)]
= [(3, 1, 5, 0, –3, 2) + (–5, 2, 3, –7, 8, 10), (7, 8, 9, 11, 0, 1) + (–2, 11, 10, –4, 5, –3)]
= [(3 + (–5), 1+2, 5+3, 0+(–7), –3 + 8, 2+10),
(7–2, 8+11, 9+10, 11+(–4) 0+5, 1 + (–3))
= [(–2, 3, 8, –7, 5, 12), (5, 19, 19, 7, 5, –2)]

is again a 1 × 6 row matrix interval.

Example 5.6: Let P = [A, B] = [(3, 7, 1), (5, 1, –3)] and

R = [C, D] = [(–2, 1, 0), (–7, 5, 2)] be two 1 × 3 row matrix intervals.

P + R = [A, B] + [C, D]
= [A + C, B + D]
= [(3, 7, 1) + (–2, 1, 0), (5, 1, –3) + (–7, 5, 2)].

Now as in case of natural class of intervals we may have A + B > C + D or A + B < C + D or A + B not comparable with C + D or A + B = C + D.

Example 5.7: Let
S = [(0, 0, 0, 0, 0, 0), (3, 1, 2, –5, 7, 2)= [S_1, S_2] and
P = [(7, 2, 1, 0, 5, 1), (0, 0, 0, 0, 0, 0)] = [P_1, P_2] be any two 1 × 6 row matrix intervals.

Now S + P = [S_1, S_2] + [P_1, P_2]
= [(S_1 + P_1), (S_2 + P_2)]
= [(0, 0, 0, 0, 0, 0) + (7, 2, 1, 0, 5, 1), (3, 1, 2, –5, 7, 2) + (0, 0, 0, 0, 0, 0)]
= [(7, 2, 1, 0, 5, 1), (3, 1, 2, –5, 7, 2)]

is again a 1 × 6 row matrix interval.

We observe that addition of 1 × n row matrix intervals is commutative for if A = $[A_1, A_2]$ and B = $[B_1, B_2]$ then A + B = B + A.

Further addition of 1 × n row intervals is associative, that is if A = $[A_1, A_2]$, B = $[B_1, B_2]$ and C = $[C_1, C_2]$ then (A + B) + C = A + (B + C).

We define 0 = [(0, 0, …, 0), (0, 0, …, 0)] as the zero interval matrix or 1 × n row zero matrix interval or 1 × n row matrix zero interval. 0 acts as the additive identity.

For if A = $[A_1, A_2]$ and (0) = [(0), (0)] then A + (0) = (0) + A = A for all A; 1 × n row matrix interval. Further if A = $[A_1, A_2]$ then –A = $[–A_1, –A_2]$ and A + (–A) = [(0), (0)].

Let V = {[A, B] = M where A and B are 1 × n row matrices with entries from R or Z or Q or Z_n or C}; V is an additive abelian group of 1 × n row matrix interval.

Now we can define product of two 1 × n row matrix intervals. Suppose A = $[A_1, B_1]$ and B = $[C_1, D_1]$ be two 1 × n row matrix intervals; we define product AB as follows:

Let AB = $[A_1, B_1]$ $[C_1, D_1]$
= $[A_1C_1, B_1D_1]$
= $[(a_1, a_2, \ldots, a_n), (c_1, c_2, \ldots, c_n)$ $(b_1, b_2, \ldots, b_n),$
$(d_1, d_2, \ldots, d_n)]$
= $[(a_1, c_1, a_2c_2, \ldots, a_nc_n), (b_1d_1, b_2d_2, \ldots, b_nd_n)]$.

We see AB is again a 1 × n row matrix interval.

We will illustrate this situation by some simple examples.

Example 5.8: Let
V = [A, B] = [(5, 7, –2, 0, 7, 8, 9, 11), (3, 1, +2, 4, 0, 18, 3, 0)] and S = $[S_1, S_2]$ = [(3, 2, 0, 1, 5, 0, –2, 0), (0, 4, 0, 2, 1, 0, 2, –4)] be two 1 × 8 row matrix intervals.

Consider VS = [A, B] [S_1, S_2]

= [AS_1, BS_2]

= [(5, 7, –2, 0, 7, 8, 9, 11), (3, 1, +2, 4, 0, 18, 3, 0),

(3, 2, 0, 1, 5, 0, –2, 0), (0, 4, 0, 2, 1, 0, 2, –4)]

= [(15, 14, 0, 0, 35, 0, –18, 0), (0, 4, 0, 8, 0, 0, 6, 0)]

is again a 1 × 8 row matrix interval.

Example 5.9: Let V = [(3, 2, 0, 5, 1), (7, 8, 1, 0, 3)] and M = [(0, 1, 2, 0, 5), (3, 0, 0, 7, 0)] be two 1 × 5 row matrix intervals. Now VM = [(3, 2, 0, 5, 1), (7, 8, 1, 0, 3), (0, 1, 2, 0, 5), (3, 0, 0, 7, 0)]

= [(0, 2, 0, 0, 5), (21, 0, 0, 0, 0)] is again a 1 × 5 row matrix interval.

Example 5.10: Let A = [(–3, 2, 5, 1, 0, 2), (7, 0, 0, 8, 0, 1)] and B = [(0, 0, 0, 0, 8, 0), (0, 7, 5, 0, –3, 0)] be two 1 × 6 row matrix intervals.

We see AB = [(–3, 2, 5, 1, 0, 2), (7, 0, 0, 8, 0, 1)]

[(0, 0, 0, 0, 8, 0), (0, 7, 5, 0, –3, 0)]

= [(–3, 2, 5, 1, 0, 2), (0, 0, 0, 0, 8, 0), (7, 0, 0, 8, 0, 1)

(0, 7, 5, 0, –3, 0)]

= [(0, 0, 0, 0, 0, 0), (0, 0, 0, 0, 0, 0)].

Thus A ≠ [(0), (0)] and B ≠ [(0), (0)],

but AB = [(0), (0)].

Now we see if V = {M = [A, B] | collection of all 1 × n row matrix intervals} then V is closed with respect to multiplication. The operation multiplication is commutative and associative on V; but we see V is only a commutative semigroup under multiplication. For we see V has infinitely many zero divisors.

Example 5.11: Let M = [A, (0)] and [(0), B] = P be two 1 × n row matrix intervals. We see MP = [(0), (0)] is a zero divisor.

Now we define subtraction of two 1 × n row matrix interval. Suppose M = [(5, 3, –1, 0, 5), (3, 2, 4, 2, 0)] and P = [(3, 0, 1, 2, 4), (2, –1, 5, 4, –3)] be two 1 × 5 row matrix intervals. We fine M – P = [(5, 3, –1, 0, 5), (3, 2, 4, 2, 0)] [(3, 0, 1, 2, 4), (2, –1, 5, 4, –3)] = [(2, 3, –2, –2, 1), (1, 3, –1, –2, 3)] is again a 1 × 5 row matrix intervals.

The following points are pertinent to be observed.

(1) The operation subtraction is not commutative.

(2) The operation of subtraction is non associative.

Consider A = [A_1, B_1] = [(3, 2, –1, 0), (8, 5, 1, 2)] and B = [A_2, B_2] = [(6, 3, 4, 8), (–4, 3, 2, 7)] two 1 × 4 row matrix intervals.

A – B = [A_1, B_1] – [A_2, B_2] = [$A_1 - A_2$, $B_1 - B_2$]
= [(3, 2, –1, 0) – (6, 3, 4, 8), (8, 5, 1, 2), (–4, 3, 2, 7)]
= [(–3, –1, –5, –8), (12, 2, –1, –5)].

Consider B – A = [A_2, B_2] – [A_1, B_1]
= [96, 3, 4, 8), (–4, 3, 2, 7)] – [(3, 2, –1, 0), (–4, 3, 2, 7)
– (8, 5, 1, 2)]
= [(3, 1, 5, 8), (–12, –2, 1, 5)].
A – B ≠ B – A.

Thus the operation subtraction is non commutative.

It is easy to check A – (B – C) = (A – B) – C for any 1 × n row matrix interval.

Consider A = (A_1, B_1), B = (A_2, B_2) and C = (A_3, B_3) be three 1 × 3 row matrix interval.

A = (A_1, B_1) = [(3, 2, 0), (–2, 1, 5)],
B = (A_2, B_2) = [(–3, –2, 4), (5, 3, –1)]
and C = (A_3, B_3) = [(2, 4, 5), (0, 1, 2)];

(A – B) – C = ([(3, 2, 0), (–2, 1, 5)] – [(–3, –2, 4), (5, 3, –1)]
– [(2, 4, 5), (0, 1, 2)]
= [(6, 4, –4), (–7, –2, 6)] – [(2, 4, 5), (0, 1, 2)]
= [(4, 0, –9), (–7, –3, 4)].

$A - (B - C) = [(3, 2, 0), (-2, 1, 5)] - ([(-3, -2, 4), (5, 3, -1)] - [(2, 4, 5), (0, 1, 2)])$
$= [(3, 2, 0), (-2, 1, 5)] - [(-5, -6, -1), (5, 2, -3)]$
$= [(8, 8, 1), (-7, -1, 8)]$.
Thus $(A - B) - C \neq A - (B - C)$.

We now proceed onto define the notion of division of two $1 \times n$ row matrix intervals.

Division of an $1 \times n$ row matrix interval $A = [A_1, B_1]$ can be divided by $B = [A_2, B_2]$ if and only if no coordinate in A_2 and B_2 is zero:

That is if $A = [A_1, B_1] = [(a_1, a_2, \ldots, a_n), (b_1, b_2, \ldots, b_n)]$ and $B = [(c_1, c_2, \ldots, c_n), (d_1, d_2, \ldots, d_n)]$.

Then $A / B = [A_1/A_2, B_1/B_2]$
$= [(a_1/c_1, a_2/c_2, \ldots, a_n/c_n), (b_1/d_1, b_2/d_2, \ldots, b_n/d_n)]$ where $c_i \neq 0$ and $d_i \neq 0$ for $1 \leq i \leq n$.

We see clearly $A/B \neq B / A$ even if none of the coordinates in A_1 and B_1 is zero.

Also the division operation is non associative.

Now we can give better structure for the $1 \times n$ row matrix interval.

THEOREM 5.1: *Let $V = \{M = [A, B]$ the collection of all $1 \times n$ row matrix intervals with entries from R or Q or Z or Z_n or C or $C(Z_n)\}$; V is a commutative ring with unit and has zero divisors.*

The proof is direct and hence is left as an exercise to the reader.

Note V in general is an infinite ring. V is a finite ring only the entries of M are from Z_n or $C(Z_n)$.

We will give one or two examples of this ring.

Example 5.12: Let
$M=\{P = [A, B] = [(a_1, a_2, a_3), (b_1, b_2, b_3)] \mid a_i, b_i \in Z_2; 1 \leq i \leq 3\}$ be the collection of all 1×3 row matrix intervals. Clearly M is a finite commutative ring with zero divisors. Further $|M| = 64$.

Example 5.13: Let
$P = \{M = [A, B] = [(a_1, a_2), (b_1, b_2)]$ where $a_i, b_i \in Z, 1 \leq i \leq 2\}$ be the collection of all 1×2 row matrix intervals P is a ring. P has no units but zero divisors. P is of infinite order.

Example 5.14: Let $K = \{M = [A, B] = [(a_1, \ldots, a_{10}), (b_1, \ldots, b_{10})] \mid a_i, b_i \in C, 1 \leq i \leq 10\}$ be a collection of all 1×10 row matrix intervals, K is a ring of infinite order.

We can define substructures on these structures. This is simple and hence is left as an exercise; these rings have subrings and ideals.

Now we bring in the representation / connection between $1 \times n$ row matrix intervals and $1 \times n$ interval matrices.

We know $N = ([a_1, b_1], \ldots, [a_n, b_n])$ is a $1 \times n$ row interval matrix and $M = [A, B]$

$= [(a_1, \ldots, a_n), (b_1, \ldots, b_n)]$ is a $1 \times n$ row matrix interval.

We see every M can be made into N and vice verse.

Consider $N = ([a_1, b_1], \ldots, [a_n, b_n]) = [(a_1, \ldots, a_n), (b_1, \ldots, b_n)]$

(by taking the first components of every interval in N together and the second components of every interval in N together.

Consider $M = [(a_1, \ldots, a_n), (b_1, \ldots, b_n)] = ([a_1, b_1], \ldots, [a_n, b_n])$.

Write M as n intervals by taking the first component of A and first component of B to form the first interval $[a_1, b_1]$ and so on we see M = [A, B] becomes N and N becomes M.

Now we proceed onto define the notion of n × 1 column matrix interval.

DEFINITION 5.3: *Let*

$$M = [A, B] = \left(\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \right)$$

be a pair of column matrices. We define M to be a m × 1 column matrix interval.

From now on wards if we say a m × 1 column matrix interval we mean a pair [A, B] where A and B are m × 1 column matrices (vectors).

We will illustrate this situation by an example.

Example 5.15: Let

$$M = \left(\begin{bmatrix} 5 \\ 3 \\ -2 \\ 0 \\ 7 \\ 8 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{bmatrix} \right),$$

M is a 7 × 1 column matrix interval.

Example 5.16: Let

$$P = \left(\begin{bmatrix} 3 \\ 0 \\ -1 \\ 2 \\ 4 \end{bmatrix}, \begin{bmatrix} 8 \\ -2 \\ 7 \\ 5 \\ 3 \end{bmatrix} \right),$$

be a 5 × 1 column matrix interval.

Example 5.17: Let

$$T = \left(\begin{bmatrix} 8 \\ 3 \\ 1 \\ 2 \\ 1 \\ 7 \end{bmatrix}, \begin{bmatrix} 0 \\ 2 \\ -4 \\ 3 \\ 0 \\ 4 \end{bmatrix} \right)$$

be a 6 × 1 column matrix interval.

We can add only two column matrix intervals of same order; just as in case of usual m × 1 column matrix interval.

We will just show how addition of two m × 1 column matrix intervals is obtained.

Example 5.18: Let

$$M = \left(\begin{bmatrix} 8 \\ 2 \\ 0 \\ 4 \\ 5 \\ 7 \\ -1 \end{bmatrix}, \begin{bmatrix} -2 \\ 0 \\ 2 \\ 5 \\ -1 \\ 8 \\ 2 \end{bmatrix} \right) \text{ and } N = \left(\begin{bmatrix} 2 \\ 1 \\ 5 \\ 4 \\ 0 \\ 3 \\ 5 \end{bmatrix}, \begin{bmatrix} -3 \\ 2 \\ -1 \\ -4 \\ 3 \\ 2 \\ 4 \end{bmatrix} \right)$$

be two 7 × 1 column matrix intervals.

$$\text{We find } M + N = \left(\begin{bmatrix} 8 \\ 2 \\ 0 \\ 4 \\ 5 \\ 7 \\ -1 \end{bmatrix}, \begin{bmatrix} -2 \\ 0 \\ 2 \\ 5 \\ -1 \\ 8 \\ 2 \end{bmatrix} \right) + \left(\begin{bmatrix} 2 \\ 1 \\ 5 \\ 4 \\ 0 \\ 3 \\ 5 \end{bmatrix}, \begin{bmatrix} -3 \\ 2 \\ -1 \\ -4 \\ 3 \\ 2 \\ 4 \end{bmatrix} \right)$$

$$= \left(\begin{bmatrix} 8 \\ 2 \\ 0 \\ 4 \\ 5 \\ 7 \\ -1 \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \\ 5 \\ 4 \\ 0 \\ 3 \\ 5 \end{bmatrix}, \begin{bmatrix} -2 \\ 0 \\ 2 \\ 5 \\ -1 \\ 8 \\ 2 \end{bmatrix} + \begin{bmatrix} -3 \\ 2 \\ -1 \\ -4 \\ 3 \\ 2 \\ 4 \end{bmatrix} \right)$$

$$= \left(\begin{bmatrix} 10 \\ 3 \\ 5 \\ 8 \\ 5 \\ 10 \\ 4 \end{bmatrix}, \begin{bmatrix} -5 \\ 2 \\ 1 \\ 1 \\ 4 \\ 10 \\ 6 \end{bmatrix} \right)$$

is again a 7 × 1 column matrix interval.

Example 5.19: Let

$$M = \left(\begin{bmatrix} 6 \\ 7 \\ 8 \\ 9 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} \right) \text{ and } P = \left(\begin{bmatrix} 5 \\ 4 \\ -3 \\ 2 \\ -1 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ 2 \\ 0 \\ 3 \end{bmatrix} \right)$$

be any two 5 × 1 column matrix intervals.

$$M + P = \left(\begin{bmatrix} 6 \\ 7 \\ 8 \\ 9 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} \right) + \left(\begin{bmatrix} 5 \\ 4 \\ -3 \\ 2 \\ -1 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ 2 \\ 0 \\ 3 \end{bmatrix} \right)$$

$$= \left(\begin{bmatrix} 6 \\ 7 \\ 8 \\ 9 \\ 0 \end{bmatrix} + \begin{bmatrix} 5 \\ 4 \\ -3 \\ 2 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \\ 2 \\ 0 \\ 3 \end{bmatrix} \right)$$

$$= \left(\begin{bmatrix} 11 \\ 11 \\ 5 \\ 11 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ 5 \\ 4 \\ 8 \end{bmatrix} \right)$$

is a 5 × 1 column matrix interval.

We see addition of two n × 1 column matrix intervals is commutative and associative.

$$0 = \left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \right)$$ is the zero column matrix interval.

Further for every

$$M = \left(\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \right) \text{ we have } -M = \left(\begin{bmatrix} -a_1 \\ -a_2 \\ \vdots \\ -a_n \end{bmatrix}, \begin{bmatrix} -b_1 \\ -b_2 \\ \vdots \\ -b_n \end{bmatrix} \right)$$

$$\text{and } M + (-M) = \left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \right).$$

Inview of all this we have the following theorem.

THEOREM 5.2: *Let*

$$V = \left\{ M = \left(\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \right) a_i, b_i \in Z \text{ or } R \text{ or } C(Z_n) \text{ or } Q \text{ or } Z_n \text{ or } C \right\}$$

be the collection of all n × 1 column matrix intervals. V is an abelian group under addition.

The proof is simple and hence is left as an exercise for the reader.

Clearly if M and N be two n × 1 column matrix intervals then MN cannot be defined or found.

Example 5.20: Let

$$M = \left\{ P = \left(\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \right) \text{ where } a_i, b_i \in Z_3, 1 \leq i \leq 4 \right\}$$

be the collection of all 4 × 1 column matrix interval. M is an abelian group under addition of finite order.

Example 5.21: Let

$$W = \left\{ S = \left(\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_7 \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_7 \end{bmatrix} \right) \text{ where } a_i, b_i \in Z, 1 \leq i \leq 7 \right\}$$

be a 7 × 1 column matrix interval group of infinite order.

Now having seen examples of m × 1 column matrix interval group. We now proceed onto define m × n matrix intervals (m ≠ n).

We will illustrate this by some examples.

$$\text{Let P} = \left(\begin{bmatrix} a_{11} & \dots & a_{1n} \\ a_{21} & \dots & a_{2n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{bmatrix}, \begin{bmatrix} b_{11} & \dots & b_{1n} \\ b_{21} & \dots & b_{2n} \\ \vdots & & \vdots \\ b_{m1} & \dots & b_{mn} \end{bmatrix}\right)$$

where a_{ij}, b_{ij} ∈ Z or Q or R or C or C (Z_n) or Z_n. 1 ≤ i ≤ m and 1 ≤ j ≤ n}. We say P is a m × n matrix interval.

We will illustrate this by some examples before we define some operations of these sets.

$$\text{Let P} = \left(\begin{bmatrix} 3 & 0 & 1 & 4 \\ 5 & 3 & 4 & 0 \\ -1 & 2 & 3 & 4 \\ 7 & 0 & 8 & 4 \\ 0 & 1 & 0 & 2 \end{bmatrix}, \begin{bmatrix} -3 & 5 & 1 & 2 \\ 0 & 1 & 2 & 0 \\ 1 & 2 & 0 & 4 \\ 5 & 1 & 6 & 2 \\ 0 & 0 & 1 & -1 \end{bmatrix}\right),$$

P is a 5 × 4 matrix interval.

We can as in case of row / column matrix intervals add any m × n matrix interval.

Consider

$$M = \left(\begin{bmatrix} 3 & 4 & 0 \\ 2 & 0 & -1 \\ 0 & 4 & -3 \\ 7 & 1 & 0 \end{bmatrix}, \begin{bmatrix} -2 & 8 & 4 \\ 0 & -1 & 2 \\ 1 & 0 & 1 \\ 6 & 4 & 0 \end{bmatrix}\right) \text{ and}$$

$$N = \left(\begin{bmatrix} -2 & 1 & 4 \\ 0 & 3 & 1 \\ 2 & 0 & 4 \\ 5 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & -2 & 3 \\ -4 & 5 & -6 \\ 7 & -8 & -9 \\ 0 & 1 & -2 \end{bmatrix} \right)$$

be two 5 × 5 matrix intervals. We find M + N as follows:

$$M + N = \left(\begin{bmatrix} 3 & 4 & 0 \\ 2 & 0 & -1 \\ 0 & 4 & -3 \\ 7 & 1 & 0 \end{bmatrix}, \begin{bmatrix} -2 & 8 & 4 \\ 0 & -1 & 2 \\ 1 & 0 & 1 \\ 6 & 4 & 0 \end{bmatrix} \right) + \left(\begin{bmatrix} -2 & 1 & 4 \\ 0 & 3 & 1 \\ 2 & 0 & 4 \\ 5 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & -2 & 3 \\ -4 & 5 & -6 \\ 7 & -8 & -9 \\ 0 & 1 & -2 \end{bmatrix} \right)$$

$$= \left(\begin{bmatrix} 3 & 4 & 0 \\ 2 & 0 & -1 \\ 0 & 4 & -3 \\ 7 & 1 & 0 \end{bmatrix} + \begin{bmatrix} -2 & 1 & 4 \\ 0 & 3 & 1 \\ 2 & 0 & 4 \\ 5 & 1 & 0 \end{bmatrix}, \begin{bmatrix} -2 & 8 & 4 \\ 0 & -1 & 2 \\ 1 & 0 & 1 \\ 6 & 4 & 0 \end{bmatrix} + \begin{bmatrix} 1 & -2 & 3 \\ -4 & 5 & -6 \\ 7 & -8 & -9 \\ 0 & 1 & -2 \end{bmatrix} \right)$$

$$= \left(\begin{bmatrix} 1 & 5 & 4 \\ 2 & 3 & 0 \\ 2 & 4 & 1 \\ 12 & 2 & 0 \end{bmatrix}, \begin{bmatrix} -1 & 6 & 7 \\ -4 & 4 & -4 \\ 8 & -8 & 10 \\ 6 & 5 & -2 \end{bmatrix} \right)$$

is again a 4 × 3 matrix interval. Thus we can add matrix intervals of same order.

In view of this we have the following theorem.

THEOREM 3.3: *Let V = {[[a_{ij}]$_{m\times n}$, [b_{ij}]$_{m\times n}$] where a_{ij}, b_{ij} $\in Z_n$ or C(Z_n) or Z or Q or R or C, $1 \leq i \leq m$, $1 \leq j \leq n$} be the $m \times n$ matrix interval. V is an additive abelian group.*

If elements of V are in Z_n or C(Z_n) then V is finite otherwise V is infinite.

However multiplication is not defined on V as multiplication is not compatible.

Example 5.22: Let

$$V = \left\{\left[\begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \\ a_5 & a_6 \end{pmatrix}, \begin{pmatrix} b_1 & b_2 \\ b_3 & b_4 \\ b_5 & b_6 \end{pmatrix}\right] \middle| \; a_i, b_j \in Z_5, 1 \leq i, j \leq 6\right\}$$

be the additive 3×2 matrix interval. V is a group of finite order and V is commutative.

Example 5.23: Let

$$M = \left\{\left[\begin{pmatrix} a_1 & a_2 & a_3 & a_4 \\ a_5 & a_6 & a_7 & a_8 \end{pmatrix}, \begin{pmatrix} b_1 & b_2 & b_3 & b_4 \\ b_5 & b_6 & b_7 & b_8 \end{pmatrix}\right] \middle| \; a_i, b_j \in Z, 1 \leq i, j \leq 8\right\}$$

be the additive 2×8 matrix group interval. M is of finite order and is commutative.

Example 5.24: Let

$$P = \left[\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \begin{pmatrix} e & f \\ g & h \end{pmatrix}\right] \text{ and } T = \left[\begin{pmatrix} a_1 & b_1 \\ c_1 & d_1 \end{pmatrix}, \begin{pmatrix} e_1 & f_1 \\ g_1 & h_1 \end{pmatrix}\right]$$

be two 2×2 matrix intervals.

We define

$$P + T = \left[\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \begin{pmatrix} e & f \\ g & h \end{pmatrix}\right] + \left[\begin{pmatrix} a_1 & b_1 \\ c_1 & d_1 \end{pmatrix}, \begin{pmatrix} e_1 & f_1 \\ g_1 & h_1 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} a+a_1 & b+b_1 \\ c+c_1 & d+d_1 \end{pmatrix}, \begin{pmatrix} e+e_1 & f+f_1 \\ g+g_1 & h+h_1 \end{pmatrix}\right]$$

is again a 2 × 2 matrix intervals.

$$PT = \left(\left[\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \begin{pmatrix} e & f \\ g & h \end{pmatrix}\right]\right)\left(\left[\begin{pmatrix} a_1 & b_1 \\ c_1 & d_1 \end{pmatrix}, \begin{pmatrix} e_1 & f_1 \\ g_1 & h_1 \end{pmatrix}\right]\right)$$

$$= \left[\begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} a_1 & b_1 \\ c_1 & d_1 \end{pmatrix}, \begin{pmatrix} e & f \\ g & h \end{pmatrix}\begin{pmatrix} e_1 & f_1 \\ g_1 & h_1 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} aa_1+bc_1 & ab_1+bd_1 \\ ca_1+dc_1 & cb_1+dd_1 \end{pmatrix}, \begin{pmatrix} ee_1+fg_1 & ef_1+fh_1 \\ ge_1+hg_1 & gf_1+hh_1 \end{pmatrix}\right]$$

is again a 2 × 2 matrix interval.

$$TP = \left[\begin{pmatrix} a_1 & b_1 \\ c_1 & d_1 \end{pmatrix}, \begin{pmatrix} e_1 & f_1 \\ g_1 & h_1 \end{pmatrix}\right]\left[\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \begin{pmatrix} e & f \\ g & h \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} aa_1+b_1c & a_1b+b_1d \\ c_1a+d_1c & bc_1+d_1d \end{pmatrix}, \begin{pmatrix} ae+bg & af+bh \\ ce+dg & cf+dh \end{pmatrix}\right].$$

Clearly PT ≠ TP.

Thus the product of two square matrix intervals is in general non commutative.

Further we see the product of square matrix intervals can also be zero.

Example 5.25: Let

$$P = \left(\begin{bmatrix} 3 & 0 & -1 \\ 2 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 2 & 0 & 1 \end{bmatrix}\right) \text{ and}$$

$$R = \left(\begin{bmatrix} 0 & 3 & 1 \\ 1 & 2 & 0 \\ 0 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 2 \\ 0 & 3 & 1 \\ 6 & 1 & 0 \end{bmatrix}\right)$$

be any two 3 × 3 matrix interval.

$$PR = \left[\begin{bmatrix} 3 & 0 & -1 \\ 2 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 2 & 0 & 1 \end{bmatrix}\right] \left[\begin{bmatrix} 0 & 3 & 1 \\ 1 & 2 & 0 \\ 0 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 2 \\ 0 & 3 & 1 \\ 6 & 1 & 0 \end{bmatrix}\right]$$

$$= \left[\begin{pmatrix} 3 & 0 & -1 \\ 2 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}\begin{pmatrix} 0 & 3 & 1 \\ 1 & 2 & 0 \\ 0 & 1 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 2 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 & 2 \\ 0 & 3 & 1 \\ 6 & 1 & 0 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 0 & 8 & 1 \\ 1 & 8 & 2 \\ 2 & 5 & 2 \end{pmatrix}\begin{pmatrix} 17 & 2 & 2 \\ 6 & 7 & 2 \\ 8 & 1 & 4 \end{pmatrix}\right]$$

is a 3 × 3 matrix interval.

Consider

$$RP = \left[\begin{pmatrix} 0 & 3 & 1 \\ 1 & 2 & 0 \\ 0 & 1 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 2 \\ 0 & 3 & 1 \\ 6 & 1 & 0 \end{pmatrix}\right] \times \left[\begin{pmatrix} 3 & 0 & -1 \\ 2 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 2 & 0 & 1 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 0 & 3 & 1 \\ 1 & 2 & 0 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 3 & 0 & -1 \\ 2 & 1 & 0 \\ 0 & 2 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 2 \\ 0 & 3 & 1 \\ 6 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 1 \\ 2 & 0 & 1 \end{pmatrix} \right]$$

$$= \left[\begin{pmatrix} 6 & 5 & 1 \\ 7 & 2 & -1 \\ 2 & 5 & 2 \end{pmatrix}, \begin{pmatrix} 5 & 0 & 4 \\ 2 & 6 & 4 \\ 6 & 2 & 13 \end{pmatrix} \right]$$

is again a 3×3 matrix interval. However RP ≠ RP.

Example 5.26: Let

$$M = \left[\begin{pmatrix} 0 & 2 & 1 & 0 \\ 4 & 0 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 3 & 4 & 0 \end{pmatrix}, \begin{pmatrix} 4 & 0 & 1 & 0 \\ 0 & 2 & 0 & 4 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \right] \text{ and}$$

$$N = \left[\begin{pmatrix} -1 & 0 & 2 & 1 \\ 0 & 1 & 0 & 2 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 & 1 \\ 2 & 0 & 1 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 2 & 0 & 1 \end{pmatrix} \right]$$

be two 4×4 matrix intervals.

$$\text{Now } N + M = \left[\begin{pmatrix} -1 & 2 & 3 & 1 \\ 4 & 1 & 0 & 3 \\ 2 & 1 & 2 & 0 \\ 0 & 4 & 4 & 1 \end{pmatrix}, \begin{pmatrix} 4 & 1 & 1 & 1 \\ 2 & 2 & 1 & 4 \\ 2 & 1 & 2 & 0 \\ 0 & 2 & 1 & 1 \end{pmatrix} \right]$$

is again a 4×4 matrix interval.

Consider MN =

$$\left[\begin{pmatrix}0&2&1&0\\4&0&0&1\\1&0&2&0\\0&3&4&0\end{pmatrix}\begin{pmatrix}-1&0&2&1\\0&1&0&2\\1&1&0&0\\0&1&0&1\end{pmatrix},\begin{pmatrix}4&0&1&0\\0&2&0&4\\1&1&0&0\\0&0&1&0\end{pmatrix}\begin{pmatrix}0&1&0&1\\2&0&1&0\\1&0&2&0\\0&2&0&1\end{pmatrix}\right]$$

$$=\left[\begin{pmatrix}1&3&0&4\\-4&1&8&5\\1&2&2&1\\4&7&0&6\end{pmatrix},\begin{pmatrix}1&4&2&4\\4&8&10&4\\2&1&1&1\\1&0&2&0\end{pmatrix}\right]$$

is a 4×4 matrix interval.

Now NM =

$$\left[\begin{pmatrix}-1&0&2&1\\0&1&0&2\\1&1&0&0\\0&1&0&1\end{pmatrix}\begin{pmatrix}0&2&1&0\\4&0&0&1\\1&0&2&0\\0&3&4&0\end{pmatrix},\begin{pmatrix}0&1&0&1\\2&0&1&0\\1&0&2&0\\0&2&0&1\end{pmatrix}\begin{pmatrix}4&0&1&0\\0&2&0&4\\1&1&0&0\\0&0&1&0\end{pmatrix}\right]$$

$$=\left[\begin{pmatrix}2&1&7&0\\4&6&8&1\\4&2&1&1\\4&3&4&1\end{pmatrix},\begin{pmatrix}0&2&1&4\\9&1&2&0\\6&2&1&0\\0&4&1&8\end{pmatrix}\right]$$

is a 4×4 matrix interval.

We see MN ≠ NM. We have the following theorem.

THEOREM 5.4: *Let $V = [[a_{ij}]_{m\times m}, [b_{ij}]_{m\times m}]$ be a collection of $m \times m$ interval matrices with a_{ij}, $b_{ij} \in Z_n$ or $C(Z_n)$ or Z or Q or R or C. V is a non commutative ring with respect to matrix interval addition and multiplication.*

This proof is direct and simple and hence left as an exercise for the reader to prove.

Clearly V is a ring with zero divisors units and idempotents.

Further we see as in case of row matrix interval we can in case of square matrix interval also obtain the square interval matrix. Also from a square interval matrix find the square matrix interval.

This we will illustrate by some examples.

Example 5.27: Let

$$V = \left[\begin{pmatrix} 5 & 0 & 1 & 2 \\ 3 & 5 & 4 & 1 \\ 2 & 3 & 0 & 4 \\ 0 & 1 & 6 & 0 \end{pmatrix}, \begin{pmatrix} 3 & -5 & 1 & 2 \\ 0 & 15 & -1 & 5 \\ 1 & 2 & 3 & 0 \\ 0 & 1 & 0 & 7 \end{pmatrix}\right]$$

be a 4 × 4 matrix interval. We have V = [V_1, V_2] = [(a_{ij}), (b_{ij})]. Now we can write V = ([a_{ij}, b_{ij}]).

$$V = \begin{pmatrix} [5,3] & [0,-5] & [1,1] & [2,2] \\ [3,0] & [5,15] & [4,-1] & [1,5] \\ [2,1] & [3,2] & [0,3] & [4,0] \\ [0,0] & [1,1] & [6,0] & [0,7] \end{pmatrix}$$

$$= \begin{pmatrix} [5,3] & [0,-5] & 1 & 2 \\ [3,0] & [5,15] & [4,-1] & [1,5] \\ [2,1] & [3,2] & [0,3] & [4,0] \\ 0 & 1 & [6,0] & [0,7] \end{pmatrix}$$

is clearly a 4 × 4 interval matrix where intervals in V are increasing or decreasing or degenerate intervals.

Thus given a 4 × 4 matrix interval we can convert it into a 4 × 4 interval matrix.

Now we just show how an interval square matrix M can be written as a square interval. Consider

$$M = \begin{bmatrix} [0,1] & [2,3] & [5,0] & [1,1] & [2,-1] \\ [4,2] & [0,0] & [7,1] & [8,0] & [0,5] \\ [1,1] & [2,4] & [3,3] & [1,-1] & [8,9] \\ [7,1] & [4,3] & [1,2] & [10,0] & [4,4] \\ [8,5] & [7,7] & [2,0] & [5,5] & [3,7] \end{bmatrix}$$

be a 5 × 5 interval matrix.

$$\text{Write } M = \left[\begin{pmatrix} 0 & 2 & 5 & 1 & 2 \\ 4 & 0 & 7 & 8 & 0 \\ 1 & 2 & 3 & 1 & 8 \\ 7 & 4 & 1 & 10 & 4 \\ 8 & 7 & 2 & 5 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 3 & 0 & 1 & -1 \\ 2 & 0 & 1 & 0 & 5 \\ 1 & 4 & 3 & -1 & 9 \\ 1 & 3 & 2 & 0 & 4 \\ 5 & 7 & 0 & 5 & 7 \end{pmatrix} \right]$$

= $[M_1, M_2]$; M is a 5 × 5 matrix interval. Thus we have the following theorem the proof of which is direct and simple.

THEOREM 3.5: *Let $M = ([m_{ij}, n_{ij}])$ be a $n \times n$ interval matrix. Then $M = [(m_{ij}), (n_{ij})]$ is a $n \times n$ matrix interval. Conversely if $A = [(a_{ij}), (b_{ij})]$ is a $n \times n$ matrix interval then $A = ([a_{ij}, b_{ij}])$ is a $n \times n$ interval matrix.*

The proof is left as an exercise to the reader.

Thus we see we can have in case of square matrix intervals the notion of determinant of intervals. If $M = [M_1, M_2]$ where M_1 and M_2 are n × n matrices then the determinant interval M is denoted by $|M| = |[M_1, M_2]| = [|M_1|, |M_2|]$.

We will first illustrate this situation by some simple examples.

Example 5.28: Let

$$P = \left[\begin{pmatrix} 3 & 4 \\ 5 & 7 \end{pmatrix}, \begin{pmatrix} 4 & 1 \\ -2 & 7 \end{pmatrix}\right]$$

be a 2 × 2 matrix interval.

Determinant interval of P denoted by

$$|P| = \left| \left[\begin{pmatrix} 3 & 4 \\ 5 & 7 \end{pmatrix}, \begin{pmatrix} 4 & 1 \\ -2 & 7 \end{pmatrix}\right] \right|$$

$$= \left[\begin{vmatrix} 3 & 4 \\ 5 & 7 \end{vmatrix}, \begin{vmatrix} 4 & 1 \\ -2 & 7 \end{vmatrix}\right]$$

= [1, 30] is again an interval matrix.

Example 5.29: Let

$$A = \left[\begin{pmatrix} 3 & 7 & 1 \\ 0 & 5 & 4 \\ 2 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 7 & 0 & 1 \\ 0 & 1 & 0 \\ 5 & 0 & -1 \end{pmatrix}\right]$$

be a 3 × 3 matrix interval.

$$|A| = \left| \left[\begin{pmatrix} 3 & 7 & 1 \\ 0 & 5 & 4 \\ 2 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 7 & 0 & 1 \\ 0 & 1 & 0 \\ 5 & 0 & -1 \end{pmatrix}\right] \right|$$

$$= \left[\begin{vmatrix} 3 & 7 & 1 \\ 0 & 5 & 4 \\ 2 & 0 & 1 \end{vmatrix}, \begin{vmatrix} 7 & 0 & 1 \\ 0 & 1 & 0 \\ 5 & 0 & -1 \end{vmatrix} \right]$$

$$= \left[3\begin{vmatrix} 5 & 4 \\ 0 & 1 \end{vmatrix} - 7\begin{vmatrix} 0 & 4 \\ 2 & 1 \end{vmatrix} + 1\begin{vmatrix} 0 & 5 \\ 2 & 0 \end{vmatrix}, \right.$$

$$\left. 7\begin{vmatrix} 1 & 0 \\ 0 & -1 \end{vmatrix} - 0\begin{vmatrix} 0 & 0 \\ 5 & -1 \end{vmatrix} + 1\begin{vmatrix} 0 & 1 \\ 5 & 0 \end{vmatrix} \right]$$

= [61, –12] is the interval determinant.

Now we show how inverse of an interval matrix can be found.

Suppose $M = [M_1, M_2]$ is a $n \times n$ matrix interval, then $M^{-1} = \left[M_1^{-1}, M_2^{-1}\right]$.

$$\text{Thus } MM^{-1} = \left[M_1M_1^{-1}, M_2M_2^{-1}\right]$$

$$= \left[M_1^{-1}M_1, M_2^{-1}M_2\right]$$

$$= M^{-1}M$$

$$= [I_n, I_n].$$

We can use any known method to determine the inverse of a $n \times n$ matrix interval.

We can use Gauss - Jordan method to find inverse for a matrix interval.

Suppose

$$M = \left(\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & & a_{nn} \end{bmatrix}, \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & & \vdots \\ b_{n1} & b_{n2} & & b_{nn} \end{bmatrix} \right)$$

= [A, B] = [(a_{ij}), (b_{ij})] then the rows of A are denoted by R_{11}, R_{12}, …, R_{1n} and the rows of B are denoted by R_{21}, R_{22}, …, R_{2n} we write M as

$$\left[\left(\begin{array}{cccc|cccc} a_{11} & a_{12} & \cdots & a_{1n} & 1 & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & a_{2n} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & & a_{nn} & 0 & 0 & \cdots & 1 \end{array} \right), \right.$$

$$\left. \left(\begin{array}{cccc|cccc} b_{11} & b_{12} & \cdots & b_{1n} & 1 & 0 & \cdots & 0 \\ b_{21} & b_{22} & \cdots & b_{2n} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ b_{n1} & b_{n2} & & b_{nn} & 0 & 0 & \cdots & 1 \end{array} \right) \right].$$

We by row operations on A and B make A and B to $I_{n\times n}$ identity matrices and all the operations are done on A and B are also simultaneously performed on $I_{n\times n}$.

The adjoined matrices $I_{n\times n}$ turns to be A^{-1} and B^{-1} respectively. That is $M^{-1} = [A^{-1}, B^{-1}]$.

For instance if

M = [A, B] = [(a_{ij}), (b_{ij})]

$$= \left[\begin{pmatrix} 3 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 5 & 2 \\ 0 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix}\right] \text{ to find } M^{-1}.$$

$$M \sim \left[\begin{pmatrix} 3 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 2 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 5 & 2 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 & 1 & 0 \\ 0 & 1 & 3 & 0 & 0 & 1 \end{pmatrix}\right]$$

$$\xrightarrow[\frac{R_{21}-R_{22}}{R_{21}}]{\frac{R_{11}-R_{13}}{R_{11}}} = \left[\left(\begin{array}{ccc|ccc} 3 & 1 & 0 & 1 & 0 & -1 \\ 0 & 1 & 2 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right), \left(\begin{array}{ccc|ccc} 1 & 5 & 0 & 1 & -1 & 0 \\ 0 & 0 & 2 & 0 & 1 & 0 \\ 0 & 1 & 3 & 0 & 0 & 1 \end{array}\right)\right]$$

$$\xrightarrow[R_{21}/2]{\frac{R_{12}-2R_{13}}{R_{12}}} = \left[\left(\begin{array}{ccc|ccc} 3 & 1 & 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 0 & 1 & -2 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right), \left(\begin{array}{ccc|ccc} 1 & 5 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 1/2 & 0 \\ 0 & 1 & 3 & 0 & 0 & 1 \end{array}\right)\right]$$

$$\xrightarrow[\frac{R_{23}-3R_{22}}{R_{23}}]{\frac{R_{11}-R_{12}}{R_{11}}} = \left[\left(\begin{array}{ccc|ccc} 3 & 0 & 0 & 1 & -1 & +1 \\ 0 & 1 & 0 & 0 & 1 & -2 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right), \left(\begin{array}{ccc|ccc} 1 & 5 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 1/2 & 0 \\ 0 & 1 & 0 & 0 & -3/2 & 1 \end{array}\right)\right]$$

$$\xrightarrow[\frac{R_{21}-5R_{23}}{R_{11}},\ R_{22}\leftrightarrow R_{23}]{R_{11}/3} = \left[\left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1/3 & -1/3 & 1/3 \\ 0 & 1 & 0 & 0 & 1 & -2 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array}\right), \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & 13/2 & -5 \\ 0 & 1 & 0 & 0 & -3/2 & 1 \\ 0 & 0 & 1 & 0 & 1/2 & 0 \end{array}\right)\right]$$

$$\text{Now } M^{-1} = \left[\begin{pmatrix} 1/3 & -1/3 & 1/3 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 13/2 & -5 \\ 0 & -3/2 & 1 \\ 0 & 1/2 & 0 \end{pmatrix}\right].$$

$$MM^{-1} = \left[\begin{pmatrix} 3 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 1/3 & -1/3 & 1/3 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 5 & 2 \\ 0 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix}\begin{pmatrix} 1 & 13/2 & -5 \\ 0 & -3/2 & 1 \\ 0 & 1/2 & 0 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\right] = [I_3, I_3].$$

Also

$$M^{-1}M = \left[\begin{pmatrix} 1/3 & -1/3 & 1/3 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} 3 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 13/2 & -5 \\ 0 & -3/2 & 1 \\ 0 & 1/2 & 0 \end{pmatrix}\begin{pmatrix} 1 & 5 & 2 \\ 0 & 0 & 2 \\ 0 & 1 & 3 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\right] = [I_3, I_3].$$

Thus $MM^{-1} = M^{-1}M$. M^{-1} is the inverse matrix interval of M.

We can using Gauss Jordan method to find the inverse matrix interval.

Now we can use the method of row reduction and bring a $m \times n$ matrix interval M to row-reduced echelon matrix interval.

(i) If the $m \times n$ matrix interval $M = [M_1, N_1]$ is row reduced

(a) If the first non zero entry in each non zero row of M_1 and N_1 is equal to 1 in M.

(b) Each column of M_1 and N_1 in M which contains the leading non zero entry of some row has all its other entries 0.

(ii) Every row of M_1 and N_1 which has all its entries 0 occurs below every row which has a non zero entry.

(iii) If rows $R_{11}, R_{12}, \ldots, R_{1s_1}$ and $R_{21}, R_{22}, \ldots, R_{2s_2}$ are non zero rows of M_1 and N_1 respectively and if the leading non zero entry of rows p and q occurs in k_p and t_q respectively of M_1 and N_1; $p = 1, 2, \ldots, s_1$ and $q = 1, 2, \ldots, s_2$ with $k_1 < k_2 < \ldots < k_{s_1}$ and $t_1 < t_2 < \ldots < t_{s_2}$.

We can also describe an m × n row reduced echelon matrix interval $M = (M_1, N_1)$ as follows:

Here $M = [(m_{ij}), (n_{ij})] = [M_1, N_1]$.

Either every entry in M_1 and N_1 is 0 or there exists positive integers s_1, s_2; $1 \leq s_1, s_2 \leq m$ and s_1 and s_2 positive integers. k_1, k_{s_1}, and $t_1, t_2, \ldots, t_{s_2}$ with $1 \leq k_i, t_j \leq n$ and

(a) $m_{ij} = 0$ and $n_{pv} = 0$ if $i > s_1$ and $m_{ij} = 0$ if $j < k_j$ and $p > s_2$ and $n_{pv} = 0$ if $v < t_v$.

(b) $m_{iki} = \delta_{ij}$; $1 \leq i \leq s_1$, $1 \leq j \leq s_1$ $n_{pt_i} = \delta_{pv}$; $1 \leq p \leq s_2$; $1 \leq v \leq s_2$.

(c) $k_1 < k_2 < \ldots < k_{s_1}$ and $t_1, t_2 < \ldots < t_{s_2}$.

We just give examples of 3 × 5 row reduced echelon matrix interval $R = [R_1, R_2]$

$$= \left[\begin{pmatrix} 0 & 1 & -2 & 0 & 1/3 \\ 0 & 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & -5 & 0 & 1/7 \\ 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}\right].$$

The following theorem is simple and hence is left as an exercise to the reader.

THEOREM 5.6: *Every m × n matrix interval $M = (M_1, N_1)$ is row equivalent to a row reduced echelon m × n matrix interval.*

P(x) is the polynomial intervals we have introduced the notion of interval polynomials in [55] and have discussed their related properties.

Suppose we have two sets of m-linear equations in n unknowns.

say

$$\begin{array}{l} A_{11} x_1 + A_{12} x_2 + \ldots + A_{1n} x_n = y_1 \\ A_{21} x_1 + A_{22} x_2 + \ldots + A_{2n} x_n = y_2 \\ \vdots \qquad \vdots \qquad\qquad \vdots \qquad \vdots \\ A_{m1}x_1 + A_{m2}x_2 + \ldots + A_{mn}x_n = y_m \end{array}$$

where $y_1 y_2 \ldots y_m$ and A_{ij}; $1 \le i \le m$, $1 \le j \le n$ are given elements of a field F.

Any n tuple $(x_1, \ldots, x_n)$ of elements of F which satisfies each of the equations is a solution of the systems.

Consider another set of m-linear equations in n-unknowns.

say

$$\begin{array}{l} B_{11}\, x_1' + B_{12}\, x_2' + \ldots + B_{1n}\, x_n' = z_1 \\ B_{21}\, x_1' + B_{22}\, x_2' + \ldots + B_{2n}\, x_n' = z_2 \\ \vdots \qquad \vdots \qquad\qquad \vdots \qquad \vdots \\ B_{m1}\, x_1' + B_{m2}\, x_2' + \ldots + B_{mn}\, x_n' = z_m \end{array}$$

where $z_1, z_2, \ldots, z_m$ and B_{ij}; $1 \le i \le m$, $1 \le j \le n$ are given elements of F.

Any n tuple $(x_1', \ldots, x_n')$ of elements of F which satisfies the equations is called the solution of the systems.

If $y_1 = y_2 = \ldots = y_m = 0$ and $z_1 = z_2 = \ldots = z_m = 0$ is homogeneous set of equations or that each of the equations is homogeneous.

We now define the notion of interval system of m linear equations in n interval unknowns.

Consider

$$\begin{bmatrix} A_{11}x_1 + ... + A_{1n}x_n & B_{11}x_1' + ... + B_{1n}x_n' \\ A_{21}x_1 + ... + A_{2n}x_n & B_{21}x_1' + ... + B_{2n}x_n' \\ \vdots \quad\quad \vdots & \vdots \quad\quad \vdots \\ A_{m1}x_1 + ... + A_{mn}x_n, & B_{m1}x_1' + ... + B_{mn}x_n' \end{bmatrix} = \left[\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}\right]$$

(5.1)

(5.1) is a system of m linear interval equations in interval n unknowns. A row matrix interval $[(x_1, x_2, \ldots, x_n), (x_1', x_2', \ldots, x_n')]$ of elements from the field F which satisfies each of equations (5.1) is called the solution interval of the system.

We will first give some examples.

Consider $\begin{bmatrix} 3x_1 - x_2 + x_3 & -x_1 + x_2 + 7x_3 \\ x_1 + 3x_3 & x_2 - 4x_3 \\ -4x_1 + 2x_2 - 5x_3 & 7x_1 - 2x_2 \end{bmatrix}$

$$= \left[\begin{bmatrix} 7 \\ 3 \\ -1 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ -8 \end{bmatrix}\right]$$

is an 3-linear interval equations in 3 unknowns.

We give yet another example.

Take

$$\begin{bmatrix} x_1 - 3x_2 + x_3 - 4x_4 + x_5 & -3x_2 + 4x_3 - x_5 \\ 8x_1 + x_2 + x_4 - 5x_5 & 2x_1 - 4x_2 + x_4 + 2x_5 \\ 7x_2 + 5x_3 - 4x_4 + 2x_5 & x_1 + 3x_3 - x_4 + 5x_5 \\ x_3 + 4x_5 & x_1 + x_2 + x_5 \end{bmatrix} = \left[\begin{bmatrix} 4 \\ 2 \\ 3 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 2 \\ 3 \\ -2 \end{bmatrix}\right]$$

this is again a 4-linear interval equations in 5 unknowns.

Yet we consider another example.

$$\begin{bmatrix} 5x_1 - x_2 + x_3 + 2x_4 & x_2 + 3x_3 - 4x_4 \\ 3x_1 + x_3 - 2x_4 & 2x_1 + x_3 + x_4 \\ 5x_2 - x_3 - 3x_4 & 3x_3 - x_2 + x_1 \\ 7x_1 + x_2 + 8x_3 & 3x_1 + x_2 - 4x_3 + x_4 \\ -2x_1 + 5x_3 - 2x_4 & 5x_2 - 3x_3 + 2x_4 \end{bmatrix} = \left[\begin{pmatrix} 3 \\ -1 \\ 2 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 1 \\ 4 \\ -5 \end{pmatrix} \right]$$

is again a 5-linear interval equations in four unknowns.

Now we can give these linear interval equations a matrix interval equations representations which is as follows.

$$\left[\begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mn} \end{pmatrix}, \begin{pmatrix} B_{11} & B_{12} & \dots & B_{1n} \\ B_{21} & B_{22} & \dots & B_{2n} \\ \vdots & \vdots & & \vdots \\ B_{m1} & B_{m2} & \dots & B_{mn} \end{pmatrix} \right]$$

$$([(x_1, \dots, x_n), (x'_1, \dots, x'_n)])$$

$$= \left[\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}, \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} \right].$$

Here $[A, B]\, [X, X'] = [Y, Z]$ where $A = (A_{ij})$, $B = (B_{ij})$

$X = (x_1, \dots, x_n)$ and $X' = (x'_1, \dots, x'_n)$.

We call [A, B] the matrix interval of coefficients of the system.

Thus this m × n matrix interval with entries from the field F is a function [A, B] from the set of pairs of integers (i, j); 1 ≤ j ≤ m; 1 ≤ j ≤ n, into the field F. The entries of the matrix interval [A, B] are scalar intervals [A(i, j), B(i, j)] = $[A_{ij}, B_{ij}]$ and quite often it is convenient to describe this matrix interval by displaying it entries in a rectangular interval array having m row intervals and n column intervals. [X, X′] is a n × 1 matrix interval and [Y, Z] is a m × 1 matrix interval.

Thus [A, B] [X, X′] = [Y, Z] or [AX, BX′] = [Y, Z] is nothing more than a shorthand notation of the system of linear interval equation.

Just like in usual matrices we can define row equivalent m × n matrix interval [A, B], [A′, B′] such that [AX, BX′] = [(0), (0)] and [A′X, BX′] = [(0), (0)] have exactly the same interval solutions (solution intervals).

The proof of this result is direct and can be derived by any interested reader.

We can as in case of usual simple matrices say a n × n matrix interval M = [A, B] is an invertible n × n matrix interval and N = [C, D] is another n × n invertible matrix interval then

(1) $M^{-1} = [A, B]^{-1} = [A^{-1}, B^{-1}]$
and $(M^{-1})^{-1} = [A^{-1} B^{-1}]^{-1}$
= [A, B] (since $(A^{-1})^{-1} = A$.
and $(B^{-1})^{-1} = B$).

If both M = [A, B] and N = [C, D] are invertible n × n matrix interval then so is MN and $(MN)^{-1} = N^{-1}M^{-1}$.

Further it can be easily verified that product of n × n invertible matrix intervals is invertible. Further if A = $[A_1, B_1]$ is invertible matrix interval; that is A row equivalent with I = $[I_n, I_n]$ the n × n identity matrix interval.

If $A = [A_1, B_1]$ is an $n \times n$ invertible matrix interval then $AX = [(0), (0)]$; that is $[A_1X_1, B_1X_2] = [(0), (0)]$ has only the trivial solution $X = [X_1, X_2] = [(0), (0)]$.

The system of interval equations $AX = Y$

$[A_1, B_1] [X_1, X_2] = [Y_1, Y_2]$ that is

$[A_1X_1, B_1X_2] = [Y_1, Y_2]$ has a solution

$X = [X_1, X_2]$ for each $n \times 1$ matrix interval $Y = [Y_1, Y_2]$.

We can define as in case of usual square matrices define these notions in case of square matrix intervals.

Let $A = [A_1, B_1]$ be a $n \times n$ matrix interval over the field F.

The principal interval minors or minor intervals of $A = [A_1, B_1] = [(A_{ij}), (B_{ij})]$; $1 \leq i, j \leq n$ are scalars

$\Delta_k(A) = [\Delta_k(A_1), \Delta_k(B_1)]$ defined by
$\Delta_k(A) = [\Delta_k A_1, \Delta_k A_2]$

$$= \left[\det \begin{bmatrix} A_{11} & \dots & A_{1k} \\ A_{21} & \dots & A_{2k} \\ \vdots & & \vdots \\ A_{k1} & \dots & A_{kk} \end{bmatrix}, \det \begin{bmatrix} B_{11} & \dots & B_{1k} \\ B_{21} & \dots & B_{2k} \\ \vdots & & \vdots \\ B_{k1} & \dots & B_{kk} \end{bmatrix} \right] \text{ where } 1 \leq k \leq n.$$

Let $A = [A_1, A_2]$ be a $n \times n$ matrix interval we say A is a $n \times n$ upper triangular matrix interval if both A_1 and A_2 are upper triangular matrix interval.

Consider $M = [M_1, N_1]$

$$= \left[\begin{pmatrix} M_{11} & M_{12} & \ldots & M_{1n} \\ 0 & M_{22} & \ldots & M_{2n} \\ 0 & 0 & \ldots & M_{3n} \\ 0 & 0 & \ldots & M_{4n} \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots & M_{nn} \end{pmatrix}, \begin{pmatrix} N_{11} & N_{12} & N_{13} & \ldots & N_{1n} \\ 0 & N_{22} & N_{23} & \ldots & N_{2n} \\ 0 & 0 & N_{33} & \ldots & N_{3n} \\ 0 & 0 & 0 & \ldots & N_{4n} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & N_{nn} \end{pmatrix}\right]$$

$$\text{that is } M = \left[\begin{pmatrix} 5 & 7 & 2 & 0 \\ 0 & 1 & -5 & -7 \\ 0 & 0 & 3 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 2 & 1 & 0 & 4 \\ 0 & 3 & 1 & -2 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 7 \end{pmatrix}\right]$$

is a 4×4 upper triangular matrix interval.

Take $A = [A_1, B_1]$ be a $n \times n$ square matrix interval. If both A_1 and B_1 are $n \times n$ lower triangular matrices then we define A to be a $n \times n$ lower triangular matrix interval.

We will just give an example before we proceed onto define further structures.

Let

$$P = [P_1, P_2] = \left[\begin{pmatrix} 5 & 2 & 0 & 0 & 1 \\ 0 & 7 & 0 & 6 & 3 \\ 0 & 0 & 5 & 2 & 1 \\ 0 & 0 & 0 & 6 & -1 \\ 0 & 0 & 0 & 0 & 8 \end{pmatrix}, \begin{pmatrix} 7 & 6 & 4 & 0 & 2 \\ 0 & 1 & 0 & 5 & 0 \\ 0 & 0 & 2 & -2 & 2 \\ 0 & 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 0 & 7 \end{pmatrix}\right]$$

is a 5×5 upper triangular matrix interval.

Consider $P^T = [P_1, P_2]^T = [P_1^T, P_2^T]$

$$= \left[\begin{pmatrix} 5 & 0 & 0 & 0 & 0 \\ 2 & 7 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 6 & 2 & 6 & 0 \\ 1 & 3 & 1 & -1 & 8 \end{pmatrix}, \begin{pmatrix} 7 & 0 & 0 & 0 & 0 \\ 6 & 1 & 0 & 0 & 0 \\ 4 & 0 & 2 & 0 & 0 \\ 0 & 5 & -2 & 1 & 0 \\ 2 & 0 & 2 & 4 & 7 \end{pmatrix} \right],$$

we see P^T is a lower triangular matrix interval.

Consider $A = [A_1, B_1]$

$$= \left[\begin{bmatrix} 6 & 0 & 0 & 0 \\ 7 & 1 & 0 & 0 \\ 8 & 2 & 4 & 0 \\ 0 & 5 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 1 & 2 & 7 & 0 \\ 3 & 0 & 1 & 5 \end{bmatrix} \right]$$

be a 4 × 4 lower triangular matrix interval.

$$A^t = [A_1^t, B_1^t] = \left[\begin{pmatrix} 6 & 7 & 8 & 0 \\ 0 & 1 & 2 & 5 \\ 0 & 0 & 4 & 1 \\ 0 & 0 & 0 & 2 \end{pmatrix}, \begin{pmatrix} 2 & 0 & 1 & 3 \\ 0 & 5 & 2 & 0 \\ 0 & 0 & 7 & 1 \\ 0 & 0 & 0 & 5 \end{pmatrix} \right]$$

is a 4 × 4 lower triangular matrix interval.

Thus we see by taking transpose we can easily covert a n × n lower triangular matrix interval into a n × n upper triangular matrix interval and vice versa.

We can easily verify the following statements are equivalent in case of an invertible n × n matrix interval $A = [A_1, A_2]$;

(a) There is an upper triangular matrix interval $P = [P_1, P_2] = \left[\left(P_{ij}^1\right), \left(P_{ij}^2\right) \right]$ with $P_{kk}^1 = P_{kk}^2 = 1$; $1 \leq k \leq n$ such that $B = AP = [A_1P_1, A_2P_2] = [B_1, B_2]$ is a lower triangular matrix interval.

(b) The principal minors of A = $[A_1, A_2]$; that is the principal minors of A_1 and A_2 are all different from 0.

We call A = $[A_1, A_2]$ to be a n × n diagonal matrix interval if A = $[A_1, A_2] = [(A_{ij}), (B_{ij})]$ than $A_{ij} = 0$ and $B_{ij} = 0$ if $i \neq j$, $1 \leq i, j \leq n$.

$$\text{Thus } [P_1, P_2] = \left[\begin{bmatrix} 7 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, \begin{bmatrix} 8 & 0 & 0 & 0 \\ 0 & 6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -7 \end{bmatrix}\right]$$

is a 4 × 4 diagonal matrix interval.

Further it can be easily verified if A = $[A_1, A_2] = [(a_{ij}), (b_{ij})]$ be a n × n square upper triangular matrix interval that is if $a_{ij} = 0$, $b_{ij} = 0$ for $i > j$, that is every entry below the main diagonal is zero. A is invertible if and only if every entry on the main diagonal of both A_1 and A_2 are different from zero.

For instance

$$A = \left[\begin{pmatrix} 8 & 0 & 4 \\ 0 & 5 & 1 \\ 0 & 0 & -8 \end{pmatrix}, \begin{pmatrix} 3 & 1 & 2 \\ 0 & 5 & 7 \\ 0 & 0 & 1 \end{pmatrix}\right]$$

the upper triangular matrix interval is invertible, where as

$$P = \left[\begin{pmatrix} 8 & 0 & 4 & 8 \\ 0 & 0 & 5 & 9 \\ 0 & 0 & 7 & 4 \\ 0 & 0 & 0 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 0 & 5 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & -7 \end{pmatrix}\right]$$

the upper triangular matrix interval is not invertible.

Prove if P = [P_1, P_2]

$$= \left[\begin{pmatrix} 1 & 1/2 & 1/3 & 1/4 \\ 1/2 & 1/3 & 1/4 & 1/5 \\ 1/3 & 1/4 & 1/5 & 1/6 \\ 1/4 & 1/5 & 1/6 & 1/7 \end{pmatrix}, \begin{pmatrix} 1 & 1/2 & 1/3 & 1/4 \\ 1/2 & 1/3 & 1/4 & 1/5 \\ 1/3 & 1/4 & 1/5 & 1/6 \\ 1/4 & 1/5 & 1/6 & 1/7 \end{pmatrix}\right]$$

is an invertible matrix interval and $P^{-1} = [P_1^{-1}, P_2^{-1}]$ has integer entries.

We can define permutation matrix interval as in case of usual matrices.

We will give an example or two.

$$P = [P_1, P_2] = \left[\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}\right]$$

is a permutation matrix interval.

Now we proceed onto give example of symmetric and skew symmetric matrix intervals.

Consider

$$H = [H_1, H_2] = \left[\begin{bmatrix} 3 & -1 & 2 & 4 \\ -1 & 8 & 4 & 0 \\ 2 & 4 & 3 & 9 \\ 4 & 0 & 9 & 2 \end{bmatrix}, \begin{bmatrix} 8 & 4 & 3 & 1 \\ 4 & -1 & 4 & 2 \\ 3 & 4 & 0 & 7 \\ 1 & 2 & 7 & 2 \end{bmatrix}\right]$$

we see in this matrix interval $a_{ij} = a_{ji}$ if $i \neq j$, $1 \leq i, j \leq 4$.

Thus if A = [A_1, B_1] = [(a_{ij}), (b_{ij})] be a square matrix interval. If in A_1 and B_1, $a_{ij} = a_{ji}$ and $b_{ij} = b_{ji}$ if $i \neq j$, $1 \leq i, j \leq n$

then we define $A = [A_1, B_1]$ to be a symmetric matrix interval. If $A = [A_1, B_1] = [(a_{ij}), (b_{ij})]$ be a square matrix interval.

If $a_{ij} = -a_{ji}$ and $b_{ij} = -b_{ji}$ for $i \neq j$ and if $i = j$ then $a_{ii} = b_{ii} = 0$, $1 \leq i \leq n$, $1 \leq i, j \leq n$, then we define A to be a skew symmetric matrix interval.

Let $P = [P_1, P_2]$

$$= \left[\begin{pmatrix} 0 & 3 & 1 & -4 & 5 & 8 \\ -3 & 0 & 2 & 1 & -3 & 0 \\ -1 & -2 & 0 & -7 & 2 & 9 \\ 4 & -1 & 7 & 0 & 5 & 1 \\ -5 & 3 & -2 & -5 & 0 & 8 \\ -8 & 0 & 9 & -1 & -8 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ -1 & 0 & -6 & -7 & -8 & -9 \\ -2 & 6 & 0 & 1 & -2 & 3 \\ -3 & 7 & -1 & 0 & 4 & -5 \\ -4 & 8 & 2 & -4 & 0 & 6 \\ -5 & 9 & -3 & 5 & -6 & 0 \end{pmatrix}\right]$$

be a 6×6 skew symmetric matrix interval.

We can as in case of usual square matrices write the square matrix interval as a sum of a square symmetric matrix interval and a square skew symmetric matrix interval.

We will illustrate this situation by an example or two.

Let

$$A = [A_1, B_1] = \left[\begin{pmatrix} 8 & 4 & 3 & 5 \\ 7 & 1 & 2 & -2 \\ 10 & -5 & 8 & 0 \\ 0 & 4 & 0 & -7 \end{pmatrix}, \begin{pmatrix} 2 & -4 & 3 & 1 \\ 0 & 5 & 7 & -2 \\ 7 & 0 & 5 & 0 \\ -2 & 4 & 0 & 8 \end{pmatrix}\right]$$

be a 4×4 square matrix interval.

$A^t = [A_1, B_1]^t =$

$$\left[A_1^t, B_1^t\right] = \left[\begin{pmatrix} 8 & 7 & 10 & 0 \\ 4 & 1 & -5 & 4 \\ 3 & 2 & 8 & 0 \\ 5 & -2 & 0 & -7 \end{pmatrix}, \begin{pmatrix} 2 & 0 & 7 & -2 \\ -4 & 5 & 0 & 4 \\ 3 & 7 & 5 & 0 \\ 1 & -2 & 0 & 8 \end{pmatrix}\right]$$

be the transpose of the 4 × 4 square matrix interval.

$$\text{Let } A = [P_1, P_2] + [Q_1, Q_2]$$

$$= [P_1 + Q_1, P_2 + Q_2]$$

$$\text{where } P_1 = \frac{A_1 + A_1^t}{2}, \; Q_1 = \frac{B_1 + B_1^t}{2},$$

$$P_2 = \frac{A_1 - A_1^t}{2} \text{ and } Q_2 = \frac{B_1 - B_1^t}{2}.$$

$$\text{Now } P_1 = \frac{\begin{pmatrix} 8 & 4 & 3 & 5 \\ 7 & 1 & 2 & -2 \\ 10 & -5 & 8 & 0 \\ 0 & 4 & 0 & -7 \end{pmatrix} + \begin{pmatrix} 8 & 7 & 10 & 0 \\ 4 & 1 & -5 & 4 \\ 3 & 2 & 8 & 0 \\ 5 & -2 & 0 & -7 \end{pmatrix}}{2}$$

$$= \frac{1}{2}\begin{pmatrix} 16 & 11 & 13 & 5 \\ 11 & 2 & -3 & 2 \\ 13 & -3 & 16 & 0 \\ 5 & 2 & 0 & -14 \end{pmatrix}.$$

Clearly P_1 is a symmetric matrix of order 4 × 4.

$$\text{Now consider } P_2 = \frac{A_1 - A_1^t}{2}$$

$$= \frac{1}{2}\left[\begin{pmatrix} 8 & 4 & 3 & 5 \\ 7 & 1 & 2 & -2 \\ 10 & -5 & 8 & 0 \\ 0 & 4 & 0 & -7 \end{pmatrix} - \begin{pmatrix} 8 & 7 & 10 & 0 \\ 4 & 1 & -5 & 4 \\ 3 & 2 & 8 & 0 \\ 5 & -2 & 0 & -7 \end{pmatrix}\right]$$

$$= \frac{1}{2}\begin{pmatrix} 0 & -3 & -7 & 5 \\ 3 & 0 & 7 & -6 \\ 7 & -7 & 0 & 0 \\ -5 & 6 & 0 & 0 \end{pmatrix}.$$

Clearly P_2 is a skew symmetric matrix of order four.

$$Q_1 = \frac{B_1 + B_1^t}{2}$$

$$= \frac{1}{2}\left[\begin{pmatrix} 2 & -4 & 3 & 1 \\ 0 & 5 & 7 & -2 \\ 7 & 0 & 5 & 0 \\ -2 & 4 & 0 & 8 \end{pmatrix} + \begin{pmatrix} 2 & 0 & 7 & -2 \\ -4 & 5 & 0 & 4 \\ 3 & 7 & 5 & 0 \\ 1 & -2 & 0 & 8 \end{pmatrix}\right]$$

$$= \frac{1}{2}\begin{pmatrix} 4 & -4 & 10 & -1 \\ -4 & 10 & 7 & 2 \\ 10 & 7 & 10 & 0 \\ -1 & 2 & 0 & 16 \end{pmatrix}.$$

It is easily verified that Q_1 is a symmetric matrix of order four.

$$Q_2 = \frac{B_1 - B_1^t}{2}$$

$$= \frac{\begin{pmatrix} 2 & -4 & 3 & 1 \\ 0 & 5 & 7 & -2 \\ 7 & 0 & 5 & 0 \\ -2 & 4 & 0 & 8 \end{pmatrix} - \begin{pmatrix} 2 & 0 & 7 & -2 \\ -4 & 5 & 0 & 4 \\ 3 & 7 & 5 & 0 \\ 1 & -2 & 0 & 8 \end{pmatrix}}{2}$$

$$= \begin{bmatrix} 0 & -4 & -4 & 3 \\ 4 & 0 & 7 & -6 \\ 4 & -7 & 0 & 0 \\ -3 & 6 & 0 & 0 \end{bmatrix}$$

is a skew symmetric matrix of order four.

$$\text{Now } A = \left[\frac{1}{2}\begin{pmatrix} 16 & 11 & 13 & 5 \\ 11 & 2 & -3 & 2 \\ 13 & -3 & 16 & 0 \\ 5 & 2 & 0 & -14 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 0 & -3 & -7 & 5 \\ 3 & 0 & 7 & -6 \\ 7 & -7 & 0 & 0 \\ -5 & 6 & 0 & 0 \end{pmatrix} \right]$$

$$\frac{1}{2}\begin{pmatrix} 4 & -4 & 10 & -1 \\ -4 & 10 & 7 & 2 \\ 10 & 7 & 10 & 0 \\ -1 & 2 & 0 & 16 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 0 & -4 & -4 & 3 \\ 4 & 0 & 7 & -6 \\ 4 & -7 & 0 & 0 \\ -3 & 6 & 0 & 0 \end{pmatrix}$$

$$= [A_1, B_1]$$

$$= [\frac{1}{2}(P_1 + P_2), \frac{1}{2}(Q_1 + Q_2)].$$

The sign column matrix interval is

$$X = [X_1, X_2] = \left[\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}, \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}\right]$$

where $a_i, b_i \in \{-1, 1\}$; $1 \le i \le n$.

$$Y = \left[\begin{pmatrix} 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{pmatrix}, \begin{pmatrix} -1 \\ 1 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \end{pmatrix}\right]$$

$= [Y_1, Y_2]$ is a sign column matrix interval or sign column vector interval.

The sign row matrix interval sign row vector interval $A = [A_1, A_2]$

$= [(a_1, \ldots, a_m), (b_1, b_2, \ldots, b_m)]$ where $a_i, b_i \in \{-1, 1\}$; $1 \le i \le m$.

Thus
$A = [(1, -1, 1, 1, 1, -1, -1, 1), (-1, 1, 1, -1, -1, 1, -1, 1)]$
is a sign row vector interval or (sign row matrix interval).

These sort of matrix intervals are useful in factor analysis.

We will give examples of orthogonal matrix interval.

Let

$$X = \left[\begin{pmatrix} 1 & -1 \\ 1 & -2 \\ 1 & 3 \end{pmatrix}, \begin{pmatrix} 2 & 1 \\ -1 & 2 \\ 0 & 1 \end{pmatrix}\right]$$

be a matrix interval.

$$X^t = \left[\begin{pmatrix} 1 & 1 & 1 \\ -1 & -2 & 3 \end{pmatrix}, \begin{pmatrix} 2 & -1 & 0 \\ 1 & 2 & 1 \end{pmatrix}\right]$$

is again a matrix interval.

Consider

$$X^t X = \left[\begin{pmatrix} 1 & 1 & 1 \\ -1 & -2 & 3 \end{pmatrix}, \begin{pmatrix} 2 & -1 & 0 \\ 1 & 2 & 1 \end{pmatrix}\right] \times \left[\begin{pmatrix} 1 & -1 \\ 1 & -2 \\ 1 & 3 \end{pmatrix}, \begin{pmatrix} 2 & 1 \\ -1 & 2 \\ 0 & 1 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 1 & 1 & 1 \\ -1 & -2 & 3 \end{pmatrix}\begin{pmatrix} 1 & -1 \\ 1 & -2 \\ 1 & 3 \end{pmatrix}, \begin{pmatrix} 2 & -1 & 0 \\ 1 & 2 & 1 \end{pmatrix}\begin{pmatrix} 2 & 1 \\ -1 & 2 \\ 0 & 1 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 3 & 0 \\ 0 & 14 \end{pmatrix}, \begin{pmatrix} 5 & 0 \\ 0 & 5 \end{pmatrix}\right].$$

Thus this is an example of a orthogonal matrix interval.

If we have X = [A, B] be a matrix interval.

$X^t = [A^t, B^t]$ be the transpose of X.

Suppose $X^tX = [A^tA, B^tB] = [I_n, I_n]$ then we call x to be a orthogonal interval matrix.

Consider

$$X = \left[\begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}\right]$$

to be the matrix interval.

$$X^t = \left[\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}\right]$$

be the transpose of the matrix interval X.

$$X^tX = \left[\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}\right] \times \left[\begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}\right]$$

$$= \left[\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\right]$$

$$= [I_2, I_2].$$

Thus X is an orthonormal matrix interval.

Suppose X = [A, B] be a matrix interval and $X^t = [A^t, B^t]$ be the transpose of the matrix interval; if $X^tX = [A^tA, B^tB] = [I_m, I_m]$ then we say X is a orthonormal matrix interval.

Interested reader can give examples of them.

This result can easily be extend to square matrix interval. If $X = [A, B]$ be a $n \times n$ square matrix interval and $X^t = [A^t, B^t]$ be the transpose of X be such that $X^tX = [I_n, I_n]$ then we define X to be a orthogonal matrix interval or more specifically X is a orthonormal matrix interval. Suppose $X^tX = [D, C]$ where D and C are just diagonal matrices or symmetric matrices then we define X to be only a orthogonal square matrix interval.

Thus we use matrix theory in a new way to study properties of interval matrices constructed using natural class of intervals.

Chapter Six

DSm Matrix of Refined Labels

[46, 52] have worked with the new notion of DSm vector spaces and DSm super vector spaces. $L_Z = \{\ldots, L_{-j}, \ldots, L_{-1}, L_0, L_1, L_2, \ldots, L_j, \ldots\} = \{L_j \mid j \in Z\}$ the set of extended labels with positive and negative indexes.

Similarly $L_Q = L_q \mid q \in Q\}$ as set of labels whose indexes are fractions. L_Q is isomorphic to Q, through the isomorphism

$$f_Q(L_q) = \frac{q}{m+1},$$

for any $q \in Q$.

On similar lines they define $L_R = \{L_j \mid j \in R\}$; R the set of real numbers.

L_R is isomorphic with R through the isomorphism

$$f_R(L_r) = \frac{r}{m+1}$$

for any $r \in R$. For more refer [46, 52].

Further they have proved $\{L_R, +, \times\}$ is a field where + is the vector addition of labels and × is the vector multiplication of labels defined as DSm field of refined labels.

If $[L_1, L_2]$ is a label interval then $L_{3/2} = L_{1.5}$ is the label in the middle of the label interval.

Also $L_{-i} = -L_i$ that occur in qualitative calculations. We just from [ ] recall the operations.

Let a, b, c ∈ R and the labels

$$L_a = \frac{a}{m+1}, \; L_b = \frac{b}{m+1} \text{ and } L_c = \frac{c}{m+1}.$$

Vector addition of labels

$$L_a + L_b = L_{a+b} = L_{\frac{a+b}{m+1}}.$$

In this chapter we just indicate how we have used matrix theory in the construction of refined label matrices. When the refined labels happen to be ordered that is

$$L_0 < L_1 < \ldots < L_m$$

m ∈ N; then we can use these matrices to built vector spaces, find eigen values and all properties pertaining to vector spaces can be derived. This has been elaborately carried out in [46, 52].

So this is an unique and a new way of using matrix theory.

Not only it is matrix of refined labels whose properties are studied, we have also invented to notion of DSm super vector spaces, which uses refined labels of super matrices.

This study is elaborately made in [46, 52].

Chapter Seven

n-MATRICES AND THEIR APPLICATIONS

In this chapter we proceed onto recall the definition, properties and applications of n-matrices. This is yet another innovative method of using matrices. For more about these concepts please refer [47, 51, 54, 56].

DEFINITION 7.1: *Let $A = A_1 \cup A_2$ where A_1 and A_2 are two matrices which are distinct and the entries are taken from Q or R or Z or Z_n or $C(Z_n)$.*

If both A_1 and A_2 are row matrices we call A to be a birow matrix. Similarly, if both A_1 and A_2 are column matrices we call A to be a bicolumn matrix; A is defined as the square bimatrix if both A_1 and A_2 are square matrices. If A_1 and A_2 are rectangular matrices we define A to be a rectangular bimatrix. If in $A = A_1 \cup A_2$ one of A_1 is a row matrix and A_2 is not a row matrix we define A to be a mixed bimatrix. That is if A_1 and A_2 are two different matrices of order $m \times n$ and $s \times t$ we call A to be mixed bimatrix.

We will illustrate this by some examples.

Example 7.1: Let

$$A = A_1 \cup A_2 = (3, 1, 5, 6, 0, 2) \cup (3, 1\ 0, 0, 5),$$

we call A to be a row bimatrix.

Example 7.2: Let

$$A = A_1 \cup A_2 = \begin{bmatrix} 3 \\ 2 \\ 1 \\ 5 \\ 6 \end{bmatrix} \cup \begin{bmatrix} 5 \\ 0 \\ -1 \\ 2 \end{bmatrix}$$

be a column bimatrix.

Example 7.3: Let

$$B = B_1 \cup B_2 = (9, 0, 1) \cup (8, 4, -4),$$

B is a row bimatrix.

Example 7.4: Let

$$A = A_1 \cup A_2 = \begin{bmatrix} 9 & 0 & 1 & 2 \\ 0 & 1 & 0 & -4 \end{bmatrix} \cup \begin{bmatrix} 4 & 2 & 7 & 1 & 0 \\ 0 & 1 & 9 & -1 & 8 \\ 5 & 6 & 8 & 1 & -4 \end{bmatrix},$$

A is a rectangular bimatrix.

Example 7.5: Let

$$A = A_1 \cup A_2 = \begin{bmatrix} 3 & 4 \\ -1 & 0 \end{bmatrix} \cup \begin{bmatrix} 8 & 0 & -1 & 2 \\ 4 & 7 & 0 & 3 \\ -1 & 2 & 0 & 1 \\ 0 & 1 & 5 & 8 \end{bmatrix}$$

be a square bimatrix.

Example 7.6: Let

$$M = (8, 0, -1, 4, 3, \sqrt{3}, 7) \cup \begin{bmatrix} 5 \\ -6 \\ 8 \\ 1 \\ -4 \end{bmatrix}$$

be a mixed bimatrix.

Example 7.7: Let

$$T = T_1 \cup T_2 = \left\{\begin{pmatrix} 9 & 0 & 2 \\ -1 & 0 & 3 \end{pmatrix}\right\} \cup \begin{bmatrix} -9 \\ 2 \\ 3 \\ 4 \\ -5 \\ 6 \end{bmatrix}$$

be a mixed bimatrix.

Example 7.8: Let

$$S = S_1 \cup S_2 = \begin{bmatrix} 9 \\ -2 \\ 3 \\ 0 \\ -3 \\ 0 \end{bmatrix} \cup \begin{bmatrix} 9 & 0 \\ -1 & 2 \end{bmatrix}$$

be the mixed bimatrix.

Example 7.9: Let

$$P = P_1 \cup P_2 = (-3, 0, 1, 2) \cup \begin{bmatrix} 3 & 1 & 2 & -1 & 5 & 7 \\ 8 & -1 & 0 & 6 & -2 & 0 \end{bmatrix}$$

be the mixed bimatrix.

Now we can have bimatrices both mixed or otherwise.

Suppose $A = A_1 \cup A_2 \cup A_3 \cup \ldots \cup A_n$; $\infty > n \geq 2$ and A_i's are distinct row matrices we define A to be n-row matrix. If n = 2 it becomes a row bimatrix. If n = 3 we get the row trimatrix and so on.

If row matrices are replaced by column matrices we call them as n-column matrix. Instead of a column matrix we use rectangular matrices we call them as n-rectangular matrix or rectangular n-matrix.

Suppose in $A = A_1 \cup A_2 \cup \ldots \cup A_n$ the A_i's are square matrices, we define A to be a n-square matrix or a square n-matrix.

We will illustrate all these situations by examples.

Example 7.10: Let $A = A_1 \cup A_2 \cup A_3 \cup A_4 \cup A_5$ = (8, –1, 0, 2, 1) ∪ (0, 5, 8) ∪ (3, 2) ∪ (–7, 1, 2, 3, 4, 5, –7) ∪ (4, 6, 8, 10, 12) be a 5-row matrix or row 5-matrix.

Example 7.11: Let

$$M = M_1 \cup M_2 \cup M_3 \cup M_4 = \begin{bmatrix} 8 \\ -1 \\ 3 \\ 0 \end{bmatrix} \cup \begin{bmatrix} -4 \\ 5 \\ 8 \\ -7 \\ 0 \\ 2 \end{bmatrix} \cup \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \\ -4 \\ 5 \\ -7 \end{bmatrix} \cup \begin{bmatrix} -8 \\ 5 \\ 0 \end{bmatrix}$$

be the column 4-matrix.

Example 7.12: Let $V = V_1 \cup V_2 \cup V_3 \cup V_4 \cup V_5 \cup V_6 \cup V_7$

$$= \begin{bmatrix} 8 & 7 \\ 0 & 1 \end{bmatrix} \cup \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 0 & 1 \end{bmatrix} \cup \begin{bmatrix} 3 & 0 \\ 0 & -5 \end{bmatrix} \cup \begin{bmatrix} 8 & 0 & 1 & 2 \\ 0 & 5 & 6 & 7 \\ 0 & 0 & 7 & 8 \\ 0 & 0 & 0 & -5 \end{bmatrix} \cup$$

$$\begin{bmatrix} 1 & -2 & 3 & -4 & 5 \\ -6 & 7 & -8 & 9 & 0 \\ -1 & 2 & -3 & 4 & -5 \\ 6 & -7 & 8 & -9 & 6 \\ 0 & 1 & 2 & 5 & 8 \end{bmatrix} \cup \begin{bmatrix} 1 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 5 & 6 \end{bmatrix} \cup \begin{bmatrix} 9 & 0 & 0 & 0 \\ 0 & 8 & 1 & 2 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 7 \end{bmatrix}$$

be a 7-square matrix or square 7-matrix.

Example 7.13: Let $T = T_1 \cup T_2 \cup T_3 \cup T_4 \cup T_5$

$$= \begin{bmatrix} 8 & 0 & 5 & 0 & -8 & -5 & 1 & 2 \\ 1 & 2 & 0 & 7 & 5 & 0 & -7 & 0 \end{bmatrix} \cup \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 9 & 1 & 2 & 8 & 3 & 7 & 5 \end{bmatrix}$$

$$\cup \begin{bmatrix} 3 & 2 \\ 1 & 5 \\ 0 & 1 \\ 4 & 2 \\ 0 & 1 \end{bmatrix} \cup \begin{bmatrix} 8 & 2 & 0 \\ -1 & 0 & 5 \\ 4 & 2 & 3 \\ 5 & 6 & 7 \\ 8 & 9 & 4 \\ 0 & 0 & 1 \\ 1 & 5 & 0 \end{bmatrix} \cup \begin{bmatrix} 8 & 0 & 1 & 2 & 0 & 1 & 4 \\ 5 & 2 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 2 & 0 & 1 & 1 & 1 \\ 6 & -1 & 0 & 5 & 0 & 8 & 7 \\ 5 & 2 & 1 & 0 & 7 & 0 & 6 \\ 6 & 0 & 2 & 6 & 1 & 1 & 0 \end{bmatrix}$$

be a rectangular 5-matrix or 5-rectangular matrix.

Example 7.14: Let $V = V_1 \cup V_2 \cup V_3 \cup V_4$

$$= (-9, 0, \sqrt{3}, I, 7+i) \cup \begin{bmatrix} 7 \\ -1 \\ 8-i \\ 5i \\ 4+8i \end{bmatrix} \cup \begin{bmatrix} 9 & -i \\ \sqrt{3} & 4 \end{bmatrix}$$

$$\cup \begin{bmatrix} 8 & -i & 7-i & 8+4i \\ 0 & 1 & 2 & 3 \\ 4i & 5-i & 0 & 0 \end{bmatrix}$$

be a 4-mixed matrix or mixed 4-matrix.

Example 7.15: Let $A = A_1 \cup A_2 \cup A_3 \cup A_4 \cup A_5 \cup A_6$

$$= (-8, 0, 1, 3+2i) \cup \begin{bmatrix} 9 & 8 \\ 8 & -i \\ 4 & 5 \\ 0 & 6 \\ 3 & -2 \end{bmatrix} \cup \begin{bmatrix} 8 & 0 & -1 & 3 & 2 & 0 \\ -1 & 6 & 0 & 2 & 4 & 0 \\ 2 & 4 & 7 & 0 & 5 & -7 \end{bmatrix}$$

$$\cup \begin{bmatrix} 3 & 8 \\ 0 & 4 \end{bmatrix} \cup \begin{bmatrix} -7 \\ i+8 \\ 7-i \\ 4+3 \\ 7+2i \end{bmatrix} \cup \begin{bmatrix} 3 & 1 & 4+i \\ 4 & 2 & 7 \\ 5 & 0 & 8-i \\ 0 & -7 & 4 \end{bmatrix}$$

be a 6-mixed matrix or mixed 6-matrix.

Now we have seen just definition and examples of them. We proceed onto give three major applications of them.

The first application of bimatrices (n-matrices) is in the construction of bilinear algebra (n-linear algebra of type I and type II). For more literature please refer [60-1].

One of the major application of these n-matrices (bimatrices is that they can be used in the construction of n-eigen values (bieigen values) and n-eigen vectors (bieigen vectors). This bulk operation helps in saving time as well as easy for calculation and comparison. Further bimatrices are used in the construction of bifuzzy models like BiFCMs, BiFRMs, n-FCMs and n-FRMs; $n \geq 2$.

We just indicate how the bieigen values (n-eigen values) are calculated. We only give illustrations for theory please refer [60-1].

Example 7.16: Let

$$M = M_1 \cup M_2 = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix} \cup \begin{bmatrix} 8 & 0 & 0 \\ 0 & -1 & 2 \\ 0 & 6 & 1 \end{bmatrix}$$

be a bimatrix associated with some linear operator. The bieigen values associated with M is as follows:

$|M - (\lambda_1 \cup \lambda_2)| = |M_1 - \lambda_1| \cup |M_2 - \lambda_2|$

$$= \left|\begin{bmatrix} 3-\lambda_1 & 0 \\ 8 & -1-\lambda_1 \end{bmatrix}\right| \cup \left|\begin{bmatrix} 8-\lambda_2 & 0 & 0 \\ 0 & -1-\lambda_2 & 2 \\ 0 & 6 & 1-\lambda_2 \end{bmatrix}\right|$$

$$= (3 - \lambda_1)(-1 - \lambda_1) \cup (8-\lambda_2)[-(1-\lambda_2^2) - 1^2] = 0 \cup 0$$
$$= -(1-\lambda_1)(3-\lambda_1) \cup -(8-\lambda_2)((1-\lambda_2^2)+1^2)$$
$$= 0 \cup 0$$

leading to $\lambda_1 = 3$ and -1 and $\lambda_2 = 8$ $\lambda_2 = \pm\sqrt{13}$.

Thus $\lambda = 3 \cup 8, 3 \cup \pm\sqrt{13}, 3 \cup -\sqrt{13}$
$-1 \cup 8, -1 \cup \sqrt{13}, -1 \cup -\sqrt{13}$.

Thus using these six bieigen values one can calculate the related bieigen vectors. By this way we can compare the eigen values and also perform the operations simultaneously.

However the entries of the matrix M are taken from the field of reals. If they are taken from the rationals all the solution will not exist. We may have partial solution in that case.

Example 7.17: Let $P = P_1 \cup P_2 \cup P_3$

$$= \begin{bmatrix} 3 & 1 \\ 0 & 5 \end{bmatrix} \cup \begin{bmatrix} 2 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix} \cup \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 9 & 0 & 3 & 0 \\ 0 & 1 & 0 & 4 \end{bmatrix}$$

be the trimatrix with the entries from reals. To find the trieigen values and trieigen vectors of P.

Consider $|P - \lambda\ (=\lambda_1 \cup \lambda_2 \cup \lambda_3)|$

$= |P_1 - \lambda_1| \cup |P_2 - \lambda_2| \cup |P_3 - \lambda_3|$

$$= \begin{vmatrix} 3-\lambda_1 & +1 \\ 0 & 5-\lambda_1 \end{vmatrix} \cup \begin{vmatrix} 2-\lambda_2 & 0 & 1 \\ 0 & 1-\lambda_2 & 0 \\ 0 & 0 & 3-\lambda_2 \end{vmatrix}$$

$$\cup \begin{vmatrix} 1-\lambda_3 & 0 & 0 & 0 \\ 0 & 2-\lambda_3 & 0 & 0 \\ 9 & 0 & 3-\lambda_3 & 0 \\ 0 & 1 & 0 & 4-\lambda_3 \end{vmatrix}$$

$$= (3-\lambda_1)(5-\lambda_1) \cup (2-\lambda_2)(1-\lambda_2) \times (3-\lambda_2) \cup (1-\lambda_3)(2-\lambda_3)(3-\lambda_3)(4-\lambda_3)$$

Thus the trieigen values are (3, 2, 1), (3, 2, 2), (3, 2, 3), (3, 2, 4), (3, 1, 1), (3, 1, 2), (3, 1, 3), (3, 1, 4), (3, 3, 1), (3, 3, 2), (3, 3, 3), (3, 3, 4), (5, 2, 1), (5, 2, 2), (5, 2, 3), (5, 2, 4), (5, 1, 2), (5, 1, 3), (5, 1, 4), (5, 1, 1), (5, 3, 1), (5, 3, 2), (5, 3, 3), (5, 3, 4).

We have 24 sets of trieigen values and their corresponding 24 set of trieigen vectors can be calculated. This is one of the major innovative method of applying the bimatrix (n-matrix) theory in bilinear algebra or (n-linear algebra).

The next set of applications of bimatrices (n-matrices) is in the fuzzy models. We just indicate how these bimatrices (n-matrices) used. We call a bimatrix $A = A_1 \cup A_2$ to be a fuzzy matrix if the entries of both A_1 and A_2 are from the unit interval [0, 1]. We make use of the min max or max min or min min or max max operations on them.

On similar lines we define n-fuzzy matrix. A matrix $S = S_1 \cup S_2 \cup \ldots \cup S_n$ $(n > 2)$ is said to be a n-fuzzy matrix if the entries of each S_i is from the unit interval [0, 1]; $1 \leq i \leq n$. We use min max or max min or other such operators on S_i's. We also at times include –1 as a fuzzy number depending on the fuzzy model. The fuzzy models which make use n-fuzzy matrices are n- FCMs, n-FRMs and n-FAMs.

For more about these structures one can refer the books [48, 54, 56].

Suppose we have a problem related with the school students.

The school student problem is dependent on the teachers also dependent on the parents and on the school management. So the school children problem is dependent on three groups. Thus we can use a fuzzy trimatrix and use the Fuzzy Relational Maps (FRMs) model to analyse the problem.

Let us assume $S_1, S_2, \ldots, S_6$ are the attributes associated with the FRM.

Let $T_1, T_2, \ldots, T_7$ be the attributes related with the teachers.

$P_1, P_2, \ldots, P_6$ are the attributes associated with the parents and $M_1, M_2, \ldots, M_8$ are the attributes associate with the management.

We get using an experts opinion a trigraph and the matrix associated with the trigraph will be the fuzzy trimatrix which serves as the dynamical system of the triFRM model. If S denotes the triFRM models tridynamical system then $S = S_1 \cup S_2 \cup S_3$

$$= \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_6 \end{array}\overset{\begin{array}{cccc} T_1 & T_2 & \ldots & T_7 \end{array}}{\left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array}\right]} \cup \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_6 \end{array}\overset{\begin{array}{cccc} P_1 & P_2 & \ldots & P_6 \end{array}}{\left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array}\right]} \cup \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_6 \end{array}\overset{\begin{array}{cccc} M_1 & M_2 & \ldots & M_8 \end{array}}{\left[\begin{array}{cccc} & & & \\ & & & \\ & & & \\ & & & \end{array}\right]}.$$

The entries of the matrices S_1, S_2 and S_3 are from the unit interval [0, 1].

Any state vector $X = X_1 \cup X_2 \cup X_3 =$

$$= (x_1^1 \ \ldots \ x_6^1) \cup (x_1^2 \ldots x_6^2) \cup (x_1^3 \ \ldots \ x_6^3)$$

where $x_i^j \in \{0, 1\}$; $1 \le i \le 6$, $1 \le j \le 3$.

We find $X \text{ o } S = X_1 \text{ o } S_1 \cup X_2 \text{ o } S_2 \cup X_3 \text{ o } S_3$
$= Y_1 \cup Y_2 \cup Y_3$
$= Y$.

where elements of Y_i are from {0, 1} Y_1 will be $(y_1^1 \ \ldots \ y_7^1)$ Y_2 will be $(y_1^2, y_2^2, \ldots, y_6^2)$ and Y_3 will be $(y_1^3, y_2^3, \ldots, y_8^3)$ $y_i^j \in \{0, 1\}$, $1 \le j \le 3$ and $1 \le i \le 8$.

Thus these n-fuzzy matrices are used in constructing n-fuzzy models, when we have n-experts or these models can also be defined as multi expert models.

Finally we see these n-matrices when constructed over Z_{p^n}, p a prime and $\infty > n \geq 1$, can be used in the construction of n-codes.

We have defined n-codes [49]. For more about the uses of n-matrices in algebraic coding theory refer [49].

The construction of n-codes using n-matrices happens to be one of the new ways of applying n-matrices to algebraic coding theory.

Thus the concept of n-matrices and their applications are innovative contributions in matrix theory by the authors.

FURTHER READING

1. Abraham, R., *Linear and Multilinear Algebra*, W. A. Benjamin Inc., 1966.

2. Albert, A., *Structure of Algebras*, Colloq. Pub., **24**, Amer. Math. Soc., 1939.

3. Asbacher, Charles, *Introduction to Neutrosophic Logic*, American Research Press, Rehoboth, 2002.

4. Birkhoff, G., and MacLane, S., *A Survey of Modern Algebra*, Macmillan Publ. Company, 1977.

5. Birkhoff, G., *On the structure of abstract algebras*, Proc. Cambridge Philos. Soc., **31** 433-435, 1995.

6. Burrow, M., *Representation Theory of Finite Groups*, Dover Publications, 1993.

7. Charles W. Curtis, *Linear Algebra – An introductory Approach*, Springer, 1984.

8. Dubreil, P., and Dubreil-Jacotin, M.L., *Lectures on Modern Algebra*, Oliver and Boyd., Edinburgh, 1967.

9. Gel'fand, I.M., *Lectures on linear algebra*, Interscience, New York, 1961.

10. Greub, W.H., *Linear Algebra*, Fourth Edition, Springer-Verlag, 1974.

11. Halmos, P.R., *Finite dimensional vector spaces*, D Van Nostrand Co, Princeton, 1958.

12. Harvey E. Rose, Linear Algebra, Bir Khauser Verlag, 2002.

13. Herstein, I.N., *Abstract Algebra,* John Wiley, 1990.

14. Herstein, I.N., *Topics in Algebra,* John Wiley, 1975.

15. Herstein, I.N., and David J. Winter, *Matrix Theory and Linear Algebra*, Maxwell Pub., 1989.

16. Hoffman, K. and Kunze, R., *Linear algebra*, Prentice Hall of India, 1991.

17. Horst P., *Matrix Algebra for social scientists,* Hot, Rinehart and Winston inc, 1963.

18. Hummel, J.A., *Introduction to vector functions,* Addison-Wesley, 1967.

19. Jacob Bill, *Linear Functions and Matrix Theory* , Springer-Verlag, 1995.

20. Jacobson, N., *Lectures in Abstract Algebra*, D Van Nostrand Co, Princeton, 1953.

21. Jacobson, N., *Structure of Rings*, Colloquium Publications, **37**, American Mathematical Society, 1956.

22. Johnson, T., *New spectral theorem for vector spaces over finite fields Z_p* , M.Sc. Dissertation, March 2003 (Guided by Dr. W.B. Vasantha Kandasamy).

23. Katsumi, N., *Fundamentals of Linear Algebra*, McGraw Hill, New York, 1966.

24. Kosko, B., *Neural Networks and Fuzzy Systems*, Prentice Hall, 2001.

25. Kostrikin, A.I, and Manin, Y. I., *Linear Algebra and Geometry*, Gordon and Breach Science Publishers, 1989.

26. Kuperman, I.B, *Approximate linear algebraic equations,* Van Nostrand Reinhold, 1971

27. Lang, S., *Algebra*, Addison Wesley, 1967.

28. Lay, D. C., *Linear Algebra and its Applications,* Addison Wesley, 2003.

29. Padilla, R., Smarandache algebraic structures, *Smarandache Notions Journal*, **9** 36-38, 1998.

30. Pettofrezzo, A. J., *Elements of Linear Algebra*, Prentice-Hall, Englewood Cliffs, NJ, 1970.

31. Pless, V.S. and Huffman W.C, *Handbook of Coding theory*, Elsevier Sciences, B.V, 1998.

32. Roman, S., *Advanced Linear Algebra*, Springer-Verlag, New York, 1992.

33. Rorres, C., and Anton H., *Applications of Linear Algebra*, John Wiley & Sons, 1977.

34. Semmes, Stephen, *Some topics pertaining to algebras of linear operators*, November 2002. http://arxiv.org/pdf/math.CA/0211171

35. Shilov, G.E., *An Introduction to the Theory of Linear Spaces*, Prentice-Hall, Englewood Cliffs, NJ, 1961.

36. Smarandache, Florentin, *Special Algebraic Structures*, in Collected Papers III, Abaddaba, Oradea, 78-81, 2000.

37. Thrall, R.M., and Tornkheim, L., *Vector spaces and matrices*, Wiley, New York, 1957.

38. Van Lint, J.H., *Introduction to coding theory*, Springer, 1999.

39. Vasantha Kandasamy, W.B., Smarandache, Florentin and K. Ilanthenral, *Introduction to bimatrices*, Hexis, Phoenix, 2005.

40. Vasantha Kandasamy, W.B., and Florentin Smarandache, *Basic Neutrosophic Algebraic Structures and their Applications to Fuzzy and Neutrosophic Models*, Hexis, Church Rock, 2005.

41. Vasantha Kandasamy, W.B., and Florentin Smarandache, *Fuzzy Cognitive Maps and Neutrosophic Cognitive Maps*, Xiquan, Phoenix, 2003.

42. Vasantha Kandasamy, W.B., *Bialgebraic structures and Smarandache bialgebraic structures*, American Research Press, Rehoboth, 2003.

43. Vasantha Kandasamy, W.B., Bivector spaces, *U. Sci. Phy. Sci.*, **11** , 186-190 1999.

44. Vasantha Kandasamy, W.B., *Linear Algebra and Smarandache Linear Algebra*, Bookman Publishing, 2003.

45. Vasantha Kandasamy, W.B. and Florentin Smarandache, *Super Linear Algebra*, Infolearnquest, Ann Arbor, 2008.

46. Vasantha Kandasamy, W.B. and Florentin Smarandache, *DSm Vector Spaces of Refined Labels*, Zip Publishing, Ohio, 2011.

47. Vasantha Kandasamy, W.B., Florentin Smarandache and K. Ilanthenral, *Introduction to Bimatrices*, Hexis, 2005.

48. Vasantha Kandasamy, W.B., Florentin Smarandache and K. Amal, *Super Fuzzy Matrices and Super Fuzzy Models for Social Scientists*, Infolearnquest, Ann Arbor, 2008.

49. Vasantha Kandasamy, W.B. and Florentin Smarandache, *New Classes of Codes for Cryptologists and Computer scientists*, Infolearnquest, Ann Arbor, 2008.

50. Vasantha Kandasamy, W.B., Florentin Smarandache and K. Ilanthenral, *Super Special Codes using Super Matrices*, Svenska fysikarkix Stockholm, Sweden, 2010.

51. Vasantha Kandasamy, W.B., Florentin Smarandache and K. Ilanthenral, *Introduction to Linear Bialgebra*, Hexis, Phoenix, 2005.

52. Vasantha Kandasamy, W.B. and Florentin Smarandache, *DSm Super vector space of Refined Labels*, Zip Publishing, Ohio, 2011.

53. Vasantha Kandasamy, W.B., Florentin Smarandache and K. Amal, *Fuzzy Linguistic Topological Spaces*, Zip Publishing, 2012.

54. Vasantha Kandasamy, W.B. and Florentin Smarandache, *Introduction to n-adaptive Fuzzy Models to Analyze Public opinion in AIDS*, Hexis, Arizona, 2006.

55. Vasantha Kandasamy, W.B. and Florentin Smarandache, *Interval Linear Algebra*, Kappa and Omega, Glendale, 2010.

56. Vasantha Kandasamy, W.B., Florentin Smarandache and K. Ilanthenral, *Elementary Fuzzy Matrix Theory and Fuzzy Models for Socio Scientists*, Automaton, 2007.

57. Vasantha Kandasamy, W.B. and Florentin Smarandache, *Exploring the extension of natural operations on intervals, matrices and complex numbers*, Zip Publishing, Ohio, 2012.

58. Vasantha Kandasamy, W.B. and Florentin Smarandache, *Algebraic Structures Using Natural Class of Intervals*, The Educational Published, Ohio, 2012.

59. Vasantha Kandasamy, W.B. and Florentin Smarandache, *Natural Product X_n on matrices*, Zip Publishing, Ohio, 2012.

60. Vasantha Kandasamy, W.B. and Florentin Smarandache, *n-linear Algebra of Type II*, Infolearnquest, Ann Arbor, 2008.

61. Vasantha Kandasamy, W.B. and Florentin Smarandache, *n-linear algebra of type I and its applications*, Infolearnquest, Ann Arbor, 2008.

62. Voyevodin, V.V., *Linear Algebra*, Mir Publishers, 1983.

63. Zadeh, L.A., Fuzzy Sets, *Inform. and control*, **8**, 338-353, 1965.

64. Zelinksy, D., *A first course in Linear Algebra*, Academic Press, 1973.

INDEX

F

L

M

N

P

R

S

ABOUT THE AUTHORS

Dr.W.B.Vasantha Kandasamy is an Associate Professor in the Department of Mathematics, Indian Institute of Technology Madras, Chennai. In the past decade she has guided 13 Ph.D. scholars in the different fields of non-associative algebras, algebraic coding theory, transportation theory, fuzzy groups, and applications of fuzzy theory of the problems faced in chemical industries and cement industries. She has to her credit 646 research papers. She has guided over 68 M.Sc. and M.Tech. projects. She has worked in collaboration projects with the Indian Space Research Organization and with the Tamil Nadu State AIDS Control Society. She is presently working on a research project funded by the Board of Research in Nuclear Sciences, Government of India. This is her 77th book.

On India's 60th Independence Day, Dr.Vasantha was conferred the Kalpana Chawla Award for Courage and Daring Enterprise by the State Government of Tamil Nadu in recognition of her sustained fight for social justice in the Indian Institute of Technology (IIT) Madras and for her contribution to mathematics. The award, instituted in the memory of Indian-American astronaut Kalpana Chawla who died aboard Space Shuttle Columbia, carried a cash prize of five lakh rupees (the highest prize-money for any Indian award) and a gold medal.

She can be contacted at vasanthakandasamy@gmail.com
Web Site: http://mat.iitm.ac.in/home/wbv/public_html/
or http://www.vasantha.in

Dr. Florentin Smarandache is a Professor of Mathematics at the University of New Mexico in USA. He published over 75 books and 200 articles and notes in mathematics, physics, philosophy, psychology, rebus, literature. In mathematics his research is in number theory, non-Euclidean geometry, synthetic geometry, algebraic structures, statistics, neutrosophic logic and set (generalizations of fuzzy logic and set respectively), neutrosophic probability (generalization of classical and imprecise probability). Also, small contributions to nuclear and particle physics, information fusion, neutrosophy (a generalization of dialectics), law of sensations and stimuli, etc. He got the 2010 Telesio-Galilei Academy of Science Gold Medal, Adjunct Professor (equivalent to Doctor Honoris Causa) of Beijing Jiaotong University in 2011, and 2011 Romanian Academy Award for Technical Science (the highest in the country). Dr. W. B. Vasantha Kandasamy and Dr. Florentin Smarandache got the 2012 New Mexico-Arizona and 2011 New Mexico Book Award for Algebraic Structures. He can be contacted at smarand@unm.edu

---

Dr. Indra Venkatbabu is Actuarial Analyst at Aon Benfield, Bengaluru Area, India. She obtained her PhD from Indian Institute of Technology, Madras in the 1998. Her thesis was based on Transportation problem.

In this book authors have given the innovative ways in which they have used matrices. Starting from ATD, RTD & CETD matrices which can be applied to any collected data depending on time. A new type of matrices called fuzzy linguistic matrices which play a major role in the construction of fuzzy linguistic models is defined. Then the super matrices are used in the construction of super linear algebra and super fuzzy models. Further interval matrices using natural class of intervals is introduced. The ingenious way of defining DSm matrices of refined labels is an interesting feature. Finally the construction of n-matrices and its use in the building of n-codes is described.

US $35.00
Educational Publisher Inc.

ISBN 9781599732084
90000 >
9 781599 732084